Modèles et algorithmes pour la simulation du contact frottant dans les matériaux complexes : application aux milieux fibreux et granulaires

Gilles Daviet

Communauté
UNIVERSITÉ Grenoble Alpes

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **Mathématiques-informatique**

Arrêté ministériel du 25 mai 2016

Présentée par

Gilles Daviet

Thèse dirigée par **Florence Bertails-Descoubes**

préparée au sein de l' **équipe-projet Bipop, Inria et Laboratoire Jean Kuntzmann** et de l'école doctorale **"Mathématiques, Sciences et Technologies de l'Information, Informatique"**

Modèles et algorithmes pour la simulation du contact frottant dans les matériaux complexes

Application aux milieux fibreux et granulaires

Thèse soutenue publiquement le **15 décembre 2016**,
devant le jury composé de :

Dr Georges-Henri Cottet
Professeur des universités, Université Grenoble Alpes, Président
Dr Robert Bridson
Adjunct Professor, University of British Columbia et
Senior Principal Researcher for Visual Effects, Autodesk, Rapporteur
Dr Ioan Ionescu
Professeur des universités, Université Paris 13, Rapporteur
Dr Jean-Marie Aubry
HDR, Visual Effects Researcher, Double Negative Visual Effects, Examinateur
Dr Pierre-Yves Lagrée
Directeur de recherche, Institut d'Alembert, CNRS, Examinateur
Dr Pierre Saramito
Directeur de recherche, Laboratoire Jean Kuntzmann, CNRS, Examinateur
Dr Florence Bertails-Descoubes
Chargée de recherche, Inria Rhône-Alpes, Directrice de thèse

Remerciements

En premier lieu, je tiens bien sûr à remercier les rapporteurs et examinateurs de cette thèse, en particulier pour leur patience quand à la lecture de ce manuscrit — qui, reconnaissons le, comporte quelques sections pouvant s'avérer arides. Je remercie également le labex PERSYVAL-Lab[1] de m'avoir octroyé la bourse m'ayant finalement permis d'effectuer cette thèse, après une demi-douzaine de refus auprès d'organismes variés — et évidemment, Florence et Pierre, pour ne pas avoir désespéré malgré lesdits refus. Merci en outre à Pierre pour m'avoir initié à la simulation des fluides complexes, et Florence, pour m'avoir recueilli tout fraîchement sorti de l'Ensimag et m'avoir donné goût à ce domaine bien spécifique de l'animation pour l'informatique graphique, puis m'avoir porté en tant qu'ingénieur de recherche et doctorant. Je remercie l'équipe Bipop et l'Inria de m'avoir accueilli près de six ans, et tous ceux, permanents ou non-permanents, que j'ai pu rencontré dans ce cadre, et qui m'ont transmis quelques intuitions quand à la mécanique du contact, l'optimisation, ou d'autres sujets moins scientifiques. Merci en particulier aux précédents doctorants de Bipop et de la Tour pour leur rôle de modèles exemplaires : Florent, dont le manuscrit aura été une source d'informations inestimable ; Alexandre, Romain, Sofia, mes co-bureaux m'ayant préparé à la Rédaction, vraisemblablement avec succès. Merci également à mes anciens collègues de Weta Digital, qui ont largement contribué à élargir mes horizons culturels et scientifiques.

Car cette thèse n'aurait pu s'achever sans les moments de détente qui l'ont entrecoupé, je remercie les InuIts (et apparentés) pour les sorties à Grenoble et ailleurs, les randonnées et autres bivouacs — et parmi eux mes illustres prédécesseurs thésards, pour m'avoir encouragé à continuer dans cette voie et pour la richesse de leurs discussions, en particulier en fin de Fam's.

Finalement, je remercie ma famille, non seulement pour son support inconditionnel mais pour avoir initié ma curiosité scientifique et mon goût pour la programmation (à travers le GW-BASIC !), sans lesquels je n'aurais sans doute jamais entrepris de telles études. Plus que tout, merci à Anaïs, qui m'aura supporté et encouragé tout au long de cette thèse, et m'aura accompagné au bout du monde.

[1] This work has been partially supported by the LabEx PERSYVAL-Lab (ANR-11-LABX-0025-01) funded by the French program Investissement d'avenir

Contents

Nomenclature

Abbreviations

ADMM Alternating Direction Method of Multipliers (Section 3.3.3)

AMA Alternating Minimization Algorithm (Section 3.3.3)

DCFP Discrete Coulomb Friction Problem (Section 2.3)

DEM Discrete-Element Modeling (Section 0.2.4)

FEM Finite-Element Modeling (Section 5.2.2)

FLIP FLuid-Implicit Particle method (Section 5.2.4)

GSM Generalized Standard Material (Section 1.2.1)

ISM Implicit Standard Material (Section 1.2.2)

MPM Material Point Method (Section 5.2.4)

NSCD Non-Smooth Contact Dynamics (Section 3.4)

PIC Particle-in-Cell method (Section 5.2.4)

SOC Second-Order Cone (Definition 1.1)

SOCP Second-Order Cone Program (Section 3.2)

SOCQP Second-Order-Cone Quadratic program (Section 2.3.2)

SOR Successive Over-Relaxations (Section 3.4.3)

DP Drucker–Prager (yield surface, Section 0.3.2)

MC Mohr–Coulomb (yield surface, Section 0.3.2

Constants

Bi Bingham number

Fr Froude number

Re Reynolds number

St Stokes number

d Dimension of the simulation space (usually 2 or 3)

α (Chapter 8 only) Scaled density difference, $\alpha := \frac{\rho_g - \rho_f}{\rho_f}$

ϵ (Chapter 8 only) Ratio of scales, $\epsilon := D_g / L$

$\boldsymbol{e}_i$ i^{th} component of the canonical basis of an Euclidean space

$\mathbb{I}_n$ Identity tensor of dimension n (n might be ommitted)

$\boldsymbol{\iota}_d$ Unit normal tensor for the scalar product $S_d^2 \to \mathbb{R}$, $\boldsymbol{\sigma}, \boldsymbol{\tau} \mapsto \frac{1}{2} \cdot \boldsymbol{\sigma} : \boldsymbol{\tau} \cdot$

$\mathfrak{s}_d$ Dimension of the space S_d of symmetric tensors with d rows and columns, $\mathfrak{s}_d = \frac{1}{2} d(d+1)$

Ω	Simulation domain (for continuum mechanics) or rigid-body
B_D	Part of the boundary of Ω with Dirichlet boundary conditions
B_N	Part of the boundary of Ω with Neumann boundary conditions
Δ_t	Timestep size
ζ	Dilatancy coefficient (in the context of granular flows)
η	Dynamic viscosity
$\boldsymbol{g}$	Gravity vector
μ	Coefficient of friction
$\boldsymbol{n}$	Normal vector (for a given contact point)
$\phi_{\max}$	Maximal volume fraction
ρ	Volumetric mass
σ_S	Shear yield stress
τ_c	Tensile yield stress
c	Cohesion coefficient
D_g	Average diameter of the grains
L	Characteristic length

Differential operators

$\nabla \cdot \boldsymbol{\tau}$	Divergence of a vector or tensor field $\boldsymbol{\tau}$
$\frac{\partial f}{\partial x}$	Partial derivative or subdifferential of a function f w.r.t. a single varibale x
$\frac{D\phi}{Dt}$	Total or *material* derivative of a field ϕ , c.f. Section 5.2.1
∇f	Gradient of a function (or of a vector or scalar field) f
$\mathrm{W}(\mathbf{v})$	Skew-symmetric part of the gradient of a vector field $\mathbf{v}$, $\mathrm{W}(\mathbf{v}) := \frac{1}{2}\nabla\mathbf{v} - \frac{1}{2}(\nabla\mathbf{v})^\intercal$
$\mathrm{D}(\mathbf{v})$	Symmetric part of the gradient of a vector field $\mathbf{v}$, $\mathrm{D}(\mathbf{v}) := \frac{1}{2}\nabla\mathbf{v} + \frac{1}{2}(\nabla\mathbf{v})^\intercal$
∂f	Subdifferential of a function f (Definition A.6)

Functions

$\mathscr{I}_C$	Characteristics function of the set C (Definition A.8)
δ^i_j	Kronecker delta
δ	Dirac delta function
f_{AC}	Alart–Curnier function, defined in Equations (1.6, 1.27)
f_{BK}	Kynch batch flux density function, defined in Section 8.1.1
f_{DS}	De Saxcé function, defined in Equation (1.29)
f_{DS}	Fischer–Burmeister function, defined in Section 4.1.1

Mathematical operators

$\langle \boldsymbol{u}, \boldsymbol{v} \rangle$	Dot product of vectors $\boldsymbol{u}$ and $\boldsymbol{v}$
$\boldsymbol{\tau} : \boldsymbol{\sigma}$	Twice-contracted tensor product. For rank-2 tensors, $\boldsymbol{\tau} : \boldsymbol{\sigma} = \sum_{i,j} \boldsymbol{\tau}_{ij}\boldsymbol{\sigma}_{ji}$
$\boldsymbol{\tau} \otimes \boldsymbol{\sigma}$	Tensor (outer) product of $\boldsymbol{\tau}$ and $\boldsymbol{\sigma}$

$\boldsymbol{u} \wedge \boldsymbol{v}$	Cross product of 3D vectors $\boldsymbol{u}$ and $\boldsymbol{v}$
Π_C	Orthogonal projection on a set (Corollary A.6)
M	Mass or stiffness matrix
W	Delassus operator
adj	Adjugate matrix (transpose of cofactor matrix)
atan2	A two-arguments arctangent function that is robust to edge cases [2]
$\langle\!\langle \cdot \rangle\!\rangle$	Average values of a field at a discontinuity
$f^\star$	Convex conjugate of a function (Definition A.7)
Conv	Convex hull of a set
Dev	Deviatoric (traceless) part of a tensor
diag	(Block-)diagonal matrix obtained by diagonal concatenation of several coefficients or blocks
dom	Effective domain of a function (Definition A.3)
epi	Epigraph of a function (Definition A.2)
Im	Image of a linear operator
relint	Relative interior of a set
$[\![\cdot]\!]$	Jump (difference in values) of a field at a discontinuity
Ker	Kernel of a linear operator
$\cdot_{\mathrm{N}}$	Normal part of a vector (w.r.t. $\boldsymbol{n}$) or tensor (w.r.t. $\boldsymbol{\iota}_d$)
$\overline{C}$	Closure of the set C
prox	Proximal operator (Definition 3.1)
Span	Set spanned by linear combinations of a set of vectors
$\cdot_{\mathrm{T}}$	Tangential part of a vector (w.r.t. $\boldsymbol{n}$) or tensor (w.r.t. $\boldsymbol{\iota}_d$)
Tr	Trace of a tensor
Bd	Boundary of a set
int	Interior of a set
$I_1(\boldsymbol{\sigma})$	First invariant of a tensor $\boldsymbol{\sigma}$, $I_1(\boldsymbol{\sigma}) := \mathrm{Tr}\,\boldsymbol{\sigma}$
$J_2(\boldsymbol{\sigma})$	Second invariant of the deviatoric part of a tensor $\boldsymbol{\sigma}$, $J_2(\boldsymbol{\sigma}) := \mathrm{Tr}\,\boldsymbol{\sigma}^2$

Sets and spaces

$\mathscr{K}_\mu$	Second-Order Cone (SOC) of aperture μ (Definition 1.1)
$\mathscr{T}_{\mu,\sigma_S}$	Truncated SOC of aperture μ and base section σ_S (Section 1.3.2)
$H^1(\Omega)$	Sobolev space $W^{1,2}$ of square integrable functions with square-integrable derivatives over a domain Ω
$H^1_0(\Omega)$	Subspace of $H^1(\Omega)$ satisfying homogeneous Dirichlet boundary conditions
$L_2(\Omega)$	Space of square integrable functions over a domain Ω
$T_h \subset L_2(\Omega)^{\mathbb{S}_d}$	Discrete space of symmetric tensor fields

[2] See https://en.wikipedia.org/wiki/Atan2

$V_h(\mathbf{0}) \subset H_0^1(\Omega)^d$ Discrete space of velocity fields

$[a, b] \subset \mathbb{R}$ Closed interval

$]a, b[\subset \mathbb{R}$ Open interval

$\mathscr{N}_C$ Normal cone to a set C (Definition A.9)

K° Polar cone to a set K (Definition A.11)

$\mathscr{C}_\mu$ Set of velocity–force solutions to the Signorini–Coulomb frictional contact law (Section 1.1.2)

$\mathscr{DP}$ Set of strain–stress solutions to the non-associated Drucker–Prager rheology (Section 1.3.3)

$\bar{\mathbb{R}}$ $\mathbb{R} \cup \{-\infty, +\infty\}$

Variables (discrete mechanics)

$\boldsymbol{\lambda}$ Lagrange multipliers associated to holonomic constraints

$\boldsymbol{q}$ Generalized coordinates

$\boldsymbol{r}$ Contact reaction forces

$\boldsymbol{u}$ Relative velocities of contacting objects

$\tilde{\boldsymbol{u}}$ Relative velocities after *de Saxcé* change of variable (Section 1.2.2)

$\boldsymbol{v}$ Generalized velocities

Variables (granular flows)

β Scaled mass field, $\beta := \alpha\phi + 1$

$\dot{\boldsymbol{\varepsilon}}$ Strain rate tensor, $\dot{\boldsymbol{\varepsilon}} := \mathrm{D}(\mathbf{u})$

$\boldsymbol{\varepsilon}$ Strain tensor

η_{eff} Effective viscosity field

$\boldsymbol{\gamma}$ Affine combination of the strain rate tensor with positive divergence

$\boldsymbol{\lambda}$ Opposite of contact stress tensor

ϕ Volume fraction field

π Product fraction field, $\pi := \phi(1-\phi)$

$\boldsymbol{\sigma}$ Stress tensor

$\mathbf{u}$ Velocity field

ξ Effective drag field

$(\mathbf{T}_j)$ Basis of the discrete space of symmetric tensors T_h

$(\mathbf{T}_j)$ Shape functions (basis scalar fields) for stresses and strains

$(\mathbf{V}_i)$ Basis of the discrete velocity space $V_h(\mathbf{0})$

(ω_i^v) Shape functions (basis scalar fields) for stresses and strains velocities

Introduction

Complex materials can be defined as large collections of discrete constituents; rigid bodies, slender elastic objects, or anything in between. In this work, we are particularly interested in the case where the interactions between the different constituents are mainly driven by *dry frictional contact*, and more specifically, where the *Coulomb friction* law holds. As the Coulomb model is a macroscopic approximation of the fine-scale interactions occurring between contacting surfaces, our study will be restricted to systems whose constituents are above a critical size, around 100µm. Moreover, we will focus on materials with no fixed structure — the different constituents are free to reorganize themselves at will, and their relative motion will only be impeded by frictional contact (and possibly cohesive) forces. Natural examples of such systems include the likes of sand and scree, but also animal fur and human hair; manufactured examples can be as diverse as dry food troves or ball (and more rarely coin) pools.

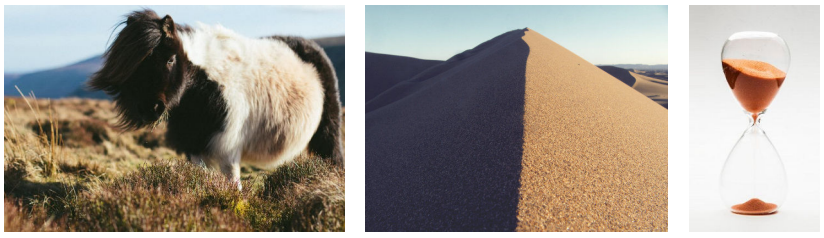
Being able to numerically reproduce the dynamics of such complex systems is important for a wide range of applications. For instance, geotechnical communities are particularly interested in the avalanching behavior of soil or gravel, while cosmetology researchers would like to assess the impact of care products on the motion of human hair. Moreover, the last decades have seen the rise of a strong demand for realism in digital special effects for feature films; the visual richness of the motion of fur, hair, or granular media have thus driven the increasing interest of the Computer Graphics community in the dynamics of complex materials.

0.1 Motivation

The numerical methods advocated in this dissertation were mostly motivated by two particular cases of complex materials, fiber assemblies and granular medias.

0.1.1 Granular materials

Granular materials (see, e.g., Andreotti et al. 2011 for a comprehensive description) commonly refer to a large collection of small solid grains larger than 100 μm in size — which typically



(a) **(b)** **(c)**

Figure 0.1: *Fur, herbs, and sand are examples of natural complex materials. The hourglass on the right illustrates the different dynamical regimes that can be exhibited by granular materials:* liquid *(above the outlet),* gaseous *(below the outlet), and* solid *(the core of the heap in the bottom compartment).*

distinguishes them from powders, made of much smaller grains. Considering this limit size, grain-grain interactions in dry granulars are mainly dictated by contact and dry friction, while air–grain interactions can be neglected. The case of immersed materials, for which interactions with the surrounding fluid can no longer be neglected, will also be treated in Chapter 8. Cohesion between grains may furthermore be considered, typically in the case of wet materials.

Being ubiquitous in outdoor environments, materials made of such grains have been heavily studied by the mechanical and geotechnical communities in the last century. They have also seen applications in a wide range of industries, including Computer Graphics. Indeed, despite their apparent simplicity, granular materials — even when constituted of rigid grains — are capable of exhibiting visually very rich dynamics. In particular, contacts and friction in such materials allow them to switch between three distinct regimes:

- a *solid* regime, when the material is maintained at rest by dry friction — for instance the core of the sand dune in Figure 0.1(b);
- a *flowing* regime, in which the material behaves like a liquid — consider the flow at the outlet of the hourglass from Figure 0.1(c), or the avalanching behavior on the outer layer of a dune;
- a *gaseous* regime, when the grains are mostly separated and only interact through sparse impacts — the flow below the hourglass' outlet, or the projections made by an impact on a granular bed.

All of these regimes (and the transitions between them) have to be properly modeled in order to produce visually convincing simulations.

0.1.2 Dynamics of hair and fur

Fibrous materials feature constituents with one dimension much longer than the other ones. Driven by industrial applications, particular cases of fibrous materials have also been the subject of extensive research. For instance, the flow of polymer suspensions is critical to injection molding, and the tire engineering community is deeply interested in the study of the cords' wear by repeated small deformations. In contrast, the large-deformation dynamics of assemblies of slender elastic rods subject to frictional contacts, as is the case of hair and fur, have historically seen less interest. Yet, industrial applications such as cosmetology and digital virtual effects have recently put the spotlight on such complex materials (Ward et al. 2007).

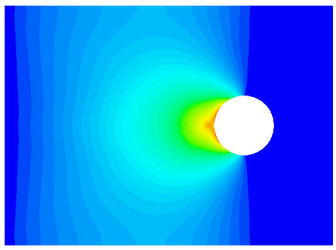
A human head of hair consists of about $150,000$ individual strands, which are very elongated, with a diameter of about 100μm for a potential length of dozens of centimeters. Conversely, animal fur such as in Figure 0.1(a) may contain millions of (generally shorter) strands. The relative importance of contact forces compared to other interactions, such as air drag or electostatic forces, is not well known. However, it is a certainty that contacts and friction play a huge role in the appearance of hair and fur, and proper handling of these interactions is of utmost importance for Computer Graphics applications. Indeed, frictional contacts maintain the volume of the groom, and thus the silhouette of the virtual character. Dry friction is furthermore responsible for the persistence of intricate patterns at rest, and ignoring it can lead to an uncannily tidy appearance. Despite these considerations, the work that we will present in Chapter 4 was among the first to attempt to properly capture dry friction in hair simulations.

0.1.3 Target applications

In this dissertation, we will not attempt to quantitatively reproduce the behavior of the simulated materials in tightly controlled settings. Instead, we will be interested in capturing the qualitative characteristics of their large-scale dynamics.

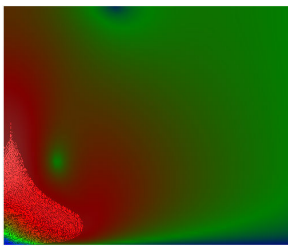
This choice is partly motivated by applications to Computer Graphics, which we will discuss in more details below. Independently of the peculiarities of this industry, we believe that convincing simulations should be based on sound physics; when possible, we will also attempt to capture in our simulations the macroscopic laws that are experimentally observed to govern the materials.

(a) *Hair dynamics, Chapter 4*



(b) *Granular pressure, Chapter 6*

(c) *Dry granular flows, Chapter 7*



(d) *Immersed avalanches, Chapter 8*

Figure 0.2: *A few snapshots of simulations from this manuscript*

Moreover, we will not solely focus on Computer Graphics; for instance, Chapters 6 and 8, in which we will study only 2D model problems, have no direct graphical application, and might be of greater interest to the mechanical engineering community.

Computer Graphics As already mentioned, a significant part of this work will be focused on devising simulation methods that are viable for Computing Graphics. A peculiarity of this application is that it strives to capture the emerging features that are created by the motion of the individual constituents. Indeed, while some of these features may not significantly influence the macroscopic mechanical properties of the material, they largely contribute to the visual richness of the overall phenomenon and would be extremely tedious to animate by hand. This overarching goal can thus be quite different from that of the mechanical engineering communities, so the simulation approaches will also be evaluated using different criteria. We list below a few of the virtues that numerical methods targeted at Computer Graphics should meet.

- *realism*, but not *accuracy*. While the human mind is good at pointing out things that "feel off", it is a poor judge about whether a simulated is physically accurate or not, especially as part of a heavily stylized movie. As such, we want to be able to capture qualitative features of our complex materials (for instance, the distinct regimes of granulars), but do not necessarily want to solve our equations to a high precision.
- *artifacts-free*. The resulting simulations must be free of disturbing visual artifacts, such as *flickering*, *popping*, *privileged directions* or *creeping*. Proper modeling of dry friction is necessary to avoid this last pitfall, and care has to be taken for the underlying discretization never to be visible.
- *controllability*. While physical realism is a good thing, being pleasant to the client's eye is a prime requirement for any Computer Graphics simulation. If a gravity-defying hair wisp is required to achieve the desired look, then the numerical method should be able to handle it. Here, we will not worry too much about this aspect. However, we will prefer models based on measurable physical parameters, and shall ensure that implementation details such as a "number of solver iterations" do not affect too much the simulated physics.
- *computational efficiency*. As providing a good set of control parameters is a hard problem (see also Sigal et al. 2015), a trial-and-error process is often required for artists to ob-

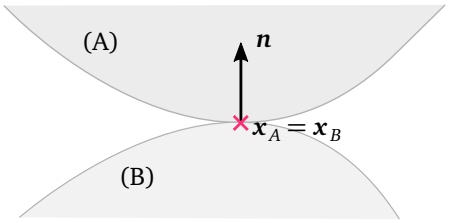


Figure 0.3: *Two bodies (A) and (B) are in contact when two points of their respective surfaces, $\boldsymbol{x}_A$ and $\boldsymbol{x}_B$, coalesce. The resulting contact normal, $\boldsymbol{n}$, is arbitrarily defined to point towards A.*

tain a satisfying look. To make this process less painful, the numerical method should be reasonably efficient, and simulations for most shots should be able to finish overnight.

0.2 Contacts and dry friction

Proper modeling of invidual constituents (for instance, in the case of fibrous materials, choosing an adequate mechanical model for individual fibers) is obviously of primary importance for computing the dynamics of any complex material. However, we will not discuss this topic in this dissertation, and will simply rely on existing models from the literature. We will instead focus on the modeling and simulation of contacts and dry friction inside the material. We provide below a brief introduction to these physical phenomena, and present the modeling choices that will underpin the remainder of this dissertation.

0.2.1 Impacts

We assume the distinct constituents of our complex material to be large enough that they can be considered to never overlap. Impacts happen when two previously disjoint objects come into contact; that is, when their (initially positive) relative distance drops to zero. Assuming sufficient smoothness of their boundaries, the two bodies will then share a normal direction along the contacting surfaces, and further relative motion of each pair of contacting points will be restricted to the half-space spanned by this normal. The simplest scenario, where two locally smooth and convex bodies (A) and (B) come into contact, is illustrated in Figure 0.3. Let $\boldsymbol{x}_A(t)$ and $\boldsymbol{x}_B(t)$ denote the position over time, for each object, of the surface point that will take part in the contact. The *gap* function, $\boldsymbol{h}(t) := \boldsymbol{x}_A(t) - \boldsymbol{x}_B(t)$, gives the relative position of those contacting points. The condition that objects (A) and (B) should not overlap can be written as $\langle \boldsymbol{h}(t), \boldsymbol{n} \rangle \geq 0$, where $\boldsymbol{n}$ is the normal to (B) at $\boldsymbol{x}_B$ and $\langle \cdot, \cdot \rangle$ denotes the usual scalar product. The two objects are in contact at instant t if $\boldsymbol{h}(t) = \boldsymbol{0}$; as long as this is the case, the normal relative velocity, $\boldsymbol{u}_{\mathrm{N}}(t) := \left\langle \frac{\mathrm{d}\boldsymbol{h}}{\mathrm{d}t}, \boldsymbol{n} \right\rangle$, should remain positive. Note that $\boldsymbol{u}$ can be discontinuous at the time of impact; however, following Moreau (1988), we will assume locally bounded variations of the relative velocity $\boldsymbol{u}$, i.e., the existence of a left-limit $\boldsymbol{u}(t^-)$ and a right-limit $\boldsymbol{u}(t^+)$ at every instant t.

At the onset of contact, a finitely-elastic body will compress, storing potential energy in the process, then restitute this energy in a second phase. Note that the amount of restituted energy does not depend on the "hardness" of the material, but rather on its internal structure; if this amount is high enough, the objects may end up separating themselves. If the elastic body is very stiff, these compression and decompression phases may happen on a time scale which is much lower than that of the studied system dynamics (Cadoux 2009, Section 1.1.1). In order to avoid having to explicitly simulate this fine time scale, one may simply model the impact as an instantaneous jump of the relative normal velocities; several models have been proposed for this purpose. The simplest of them, the empirical Newton impact law, simply states that the post-impact normal velocity should be opposed and proportional to the pre-impact one, with a

material-dependent *restitution* coefficient. On our simple example, this means that $\boldsymbol{u}_{\mathrm{N}}(t_i^+) = -\xi \boldsymbol{u}_{\mathrm{N}}(t_i^-) \geq 0$, t_i denotes the time of impact and where $0 \leq \xi \leq 1$ is the restitution coefficient. Note that this naive law may yield incorrect results in the presence of simultaneous contacts, and for instance will fail to reproduce the alternation in the contact points of a rigid block rocking on the ground; see (Brogliato 1999) for more discussion about impact laws.

In this work, we will focus on the simplest case of purely inelastic impacts — the energy will be instantaneously dissipated by the system. We shall thus enforce the post-impact normal velocity, $\boldsymbol{u}_{\mathrm{N}}(t_i^+)$, to always be null, and refer to (Cadoux 2009, Section 1.1.5) and (Smith et al. 2012) for suggestions about how the numerical framework used throughout this dissertation may be adapted to handle Newton-like impacts.

Signorini conditions Physically, the inter-penetration of the two objects is prevented by the onset of a contact-force, $\boldsymbol{r}$. In the absence of friction and cohesion, this force should be colinear to the contact normal, i.e., $\boldsymbol{r} = \boldsymbol{r}_{\mathrm{N}}\boldsymbol{n}$ with $\boldsymbol{r}_{\mathrm{N}} \geq 0$. Now, suppose that the two objects are in contact at time t, i.e., $\boldsymbol{h}_{\mathrm{N}}(t) = 0$. We have already stated the following conditions:

1. The normal relative velocity should be positive as long as the points are in contact, i.e., $\boldsymbol{h}_{\mathrm{N}} = 0 \implies \boldsymbol{u}_{\mathrm{N}} \geq 0$.
2. If t is a time of impact, the post-impact normal velocity, $\boldsymbol{u}_{\mathrm{N}}(t^+)$, should vanish.
3. The normal contact force should be positive, and vanish when $\boldsymbol{h}_{\mathrm{N}} > 0$.

The Signorini conditions are constructed by considering each contact in isolation, or more precisely, by assuming that the shock due to an impact does not propagate to other contacts. This is consistent with our choice of a purely inelastic impact law, which already forbids energy restitution. In the more general setting of a Newton impact law, this means that the pre-impact velocities are computed independently for each contact point, or again that the work of the normal contact force at already existing contacts should not be strictly positive, i.e., $\boldsymbol{u}_{\mathrm{N}}(t^-) = 0 \implies \boldsymbol{u}_{\mathrm{N}}(t^+)\boldsymbol{r}_{\mathrm{N}}(t^+) \leq 0$. Another characterization of this hypothesis is that if t is not a time of impact for the considered contact, then the contact force should locally be of bounded variations at t, i.e., should possess both left and right limits. Note that this strategy is unable to correctly model the famous Newton craddle, as the impacted balls would be incorrectly predicted to stick together (see Smith et al. 2012, Figure 3, bottom).

Combining this new implication with the three previous ones, we get that

$$\begin{cases} \boldsymbol{u}_{\mathrm{N}}(t^+) \geq 0 & \\ \boldsymbol{r}_{\mathrm{N}}(t^+) = 0 & \text{if } \boldsymbol{u}_{\mathrm{N}}(t^+) > 0 \\ \boldsymbol{r}_{\mathrm{N}}(t^+) \geq 0 & \text{if } \boldsymbol{u}_{\mathrm{N}}(t^+) = 0 \end{cases} \tag{1}$$

which together are known as the *Signorini* conditions. Remember however that these velocity-level conditions apply only when the objects are in contact at instant t, i.e., when $\boldsymbol{h}_{\mathrm{N}}(t) = 0$.

The Signorini conditions are more commonly written in a more compact manner, making the use of complementarity notation and dropping the time variable, as

$$0 \leq \boldsymbol{u}_{\mathrm{N}} \perp \boldsymbol{r}_{\mathrm{N}} \geq 0,$$

where the $\boldsymbol{u}_{\mathrm{N}} \perp \boldsymbol{r}_{\mathrm{N}}$ notation means that the two variables should be orthogonal, i.e., $\boldsymbol{r}_{\mathrm{N}}\, \boldsymbol{u}_{\mathrm{N}} = 0$.

0.2.2 Dry friction

The set of inequalities commonly referred to as the Coulomb friction law is actually the result of observations made by several authors and over the span of many centuries (Besson 2007). The discovery of the proportionality between the maximal tangential friction force and the normal load, as well as the irrelevance of the apparent contact surface area, are attributed to Leonard de Vinci at the end of the fifteenth century, and independently to Guillaume Amontons two hundred years later. Leonhard Euler distinguishes the *static* (or sticking) regime, when there is

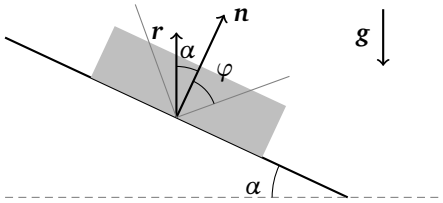


Figure 0.4: *Euler observed that as long as the angle α of an inclined plane remains below the friction angle $\varphi = \arctan \mu_S$, the ratio of the tangential to normal components of the contact force $\boldsymbol{r}$ will remain below μ_S, and the body (gray) cannot not slide.*

no relative motion between the two objects, from the dynamic regime, when one object is sliding on top of the other. Euler also introduced the notation μ_S for the static *friction coefficient*, i.e., the maximum ratio between the tangential and normal components of the reaction force, and related this coefficient to the maximum angle φ at which a mass may rest on an inclined plane without sliding as $\mu_S = \tan\varphi$ (see Figure 0.4).

In his famed manuscript, *Théorie des machines simples: en ayant égard au frottement de leurs parties et à la roideur des cordages*, Coulomb (1781) compiled the results of several experiments, validating previous theories and noting that in the dynamic regime, the friction coefficient was independent of the sliding velocity. He also observed that for most materials, the static friction coefficient, μ_S, was higher than the dynamic one, μ_D. Overall, Coulomb observed the relationship between the normal and tangential forces as obeying

$$\begin{cases} \|\boldsymbol{r}_{\mathrm{T}}\| \leq \mu_S \boldsymbol{r}_{\mathrm{N}} & \text{if } \boldsymbol{u}_{\mathrm{T}} = \boldsymbol{0} \quad \text{(static regime)} \\ \|\boldsymbol{r}_{\mathrm{T}}\| = \mu_D \boldsymbol{r}_{\mathrm{N}} & \text{if } \boldsymbol{u}_{\mathrm{T}} \neq \boldsymbol{0} \quad \text{(dynamic regime),} \end{cases} \tag{2}$$

where the $\cdot_{\mathrm{T}}$ denotes the tangential part of the reaction force and relative velocity vectors, e.g., $\boldsymbol{r}_{\mathrm{T}} = \boldsymbol{r} - \boldsymbol{r}_{\mathrm{N}}\boldsymbol{n}$. Incidentally, the proportionality of friction to the applied load was initially postulated by Amontons to be the result of the upper object having to elevate itself above the fine-scale irregularities of the contact surface. However, investigations in the twentieth century showed that this relationship is actually caused by an increase of the microscopic-level contact area when a higher normal load is applied (Bowden and Tabor 1950).

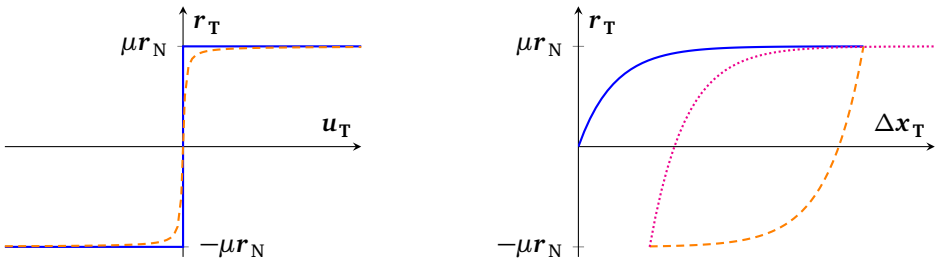
In this work, we will not distinguish between the static and dynamic friction coefficients. Indeed, the main characteristics of Coulomb friction, such as the existence of a sliding threshold that depends on the applied normal load, can already be captured without making this distinction, and we did not judge the gain in realism brought by the introduction of a distinct sliding friction coefficient worth the significant associated increase in mathematical complexity[3]. Taking into account the fact that the tangential friction force must oppose the sliding velocity, we will thus consider the Coulomb friction law as defined by the disjunction (3),

$$\begin{cases} \|\boldsymbol{r}_{\mathrm{T}}\| \leq \mu \boldsymbol{r}_{\mathrm{N}} & \text{if } \boldsymbol{u}_{\mathrm{T}} = \boldsymbol{0} \\ \begin{cases} \|\boldsymbol{r}_{\mathrm{T}}\| = \mu \boldsymbol{r}_{\mathrm{N}} \\ \boldsymbol{r}_{\mathrm{T}} = -\alpha \boldsymbol{u}_{\mathrm{T}},\ \alpha \in \mathbb{R}_+ \end{cases} & \text{if } \boldsymbol{u}_{\mathrm{T}} \neq \boldsymbol{0}. \end{cases} \tag{3}$$

0.2.3 Other friction laws

While our work will be focused on Coulomb friction, we mention below a few other friction laws that are worthy of interest.

[3] In discrete-time numerical algorithms, the friction coefficient can always be updated explicitly at each timestep depending on the status of each contact point.



(a) *Multi-valued (solid, blue) and regularized (dashed, orange) 2D friction laws*

(b) *Hysteresis cycles modeled by Dahl's friction law. Δx_T denotes the relative tangential displacement, $\Delta x_T = \int u_T dt$*

Figure 0.5: *Some alternative friction laws: regularized (left) and Dahl's (right)*

Regularization In contrast to a fluid (or *viscous*) friction law, which could be defined as, say, $\boldsymbol{r}_T = -\eta(\boldsymbol{u})\boldsymbol{u}_T$, the Coulomb friction law (3) is *multi-valued*. For $\boldsymbol{u} = \boldsymbol{0}$, the friction force is not uniquely defined, but may lie anywhere inside a ball of radius μr_N. As such, dealing with Coulomb friction will require devising specialized numerical method (Acary and Brogliato 2008). To avoid this complexity, one may choose to *regularize* the law, writing $\boldsymbol{r}_T = -\alpha(\boldsymbol{u}_T, r_N)\boldsymbol{u}_T$ with, for instance, $\alpha(\boldsymbol{u}_T, r_N) := \min\left(\mu r_N, \frac{1}{\epsilon}\|\boldsymbol{u}_T\|\right)/\|\boldsymbol{u}_T\|$, or $\alpha(\boldsymbol{u}_T, r_N) := \mu r_N \frac{2}{\pi}\left(\arctan\frac{1}{\epsilon}\|\boldsymbol{u}_T\|\right)/\|\boldsymbol{u}_T\|$ with ϵ small, as in Figure 0.5(a). However, such approaches may result in very stiff numerical systems, prone to flickering, and allow the lingering of creeping residual velocities; we will thus avoid this strategy.

Tresca model The Tresca friction law can be derived from the Coulomb law by removing the dependency of the friction force on the normal applied load,

$$\begin{cases} \|\boldsymbol{r}_T\| \leq s & \text{if } \boldsymbol{u}_T = \boldsymbol{0} \\ \begin{cases} \|\boldsymbol{r}_T\| = s \\ \boldsymbol{r}_T = -\alpha \boldsymbol{u}_T,\ \alpha \in \mathbb{R}_+ \end{cases} & \text{if } \boldsymbol{u}_T \neq \boldsymbol{0}, \end{cases}$$

where s is a positive scalar. As it reduces the coupling between tangential and normal components, the Tresca law is easier to handle numerically, yet still models a proper sliding threshold. However, the lack of proportionality of the frictional force to the applied load forbids the modeling of arbitrarily-sized stable heaps using Tresca friction, and we will thus discard this approximation.

Dahl model Taking inspiration from standard strain–stress diagrams, the Dahl (1968) friction model describes the evolution of the friction force w.r.t. the relative tangential displacement, with a slope "reset" at each change of sign of the tangential velocity $\boldsymbol{u}_T$. As depicted in Figure 0.5(b), this model is able to capture hysteresis cycles induced by friction in loading–unloading experiments, with a smooth reversing of the friction force, and has been especially popular to macroscopically account for the displacement of textile fibers under stretching and bending (Miguel et al. 2013; Ngo Ngoc and Boivin 2004). This smooth reversing of the friction force models slack in the sticking contact regime, and is therefore not really relevant for contacts between stiff bodies, which are close to slackless; we will thus not consider Dahl's law in the following.

Rolling friction Rolling friction is induced by the deformation of a wheel near its contact point with the ground, and is responsible for energy dissipation that is not captured by the tangential Coulomb friction law (3). Indeed, Equation (3) implies that the work of the friction force is non-zero only when sliding occurs. However, in the remainder of this dissertation we will focus on stiff materials and relatively small applied loads, and will thus neglect rolling friction.

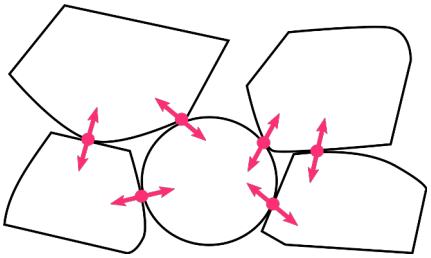


Figure 0.6: *With Discrete Element Modeling, constituents are simulated individually, and all interactions between neighboring bodies must be taken into account.*

0.2.4 Discrete simulation of complex materials with frictional contacts

The most natural way to simulate complex materials numerically would be to follow the framework of Discrete Element Modeling (DEM), [4] that is, simulating individually each body and its interactions with the surrounding ones, as illustrated in Figure 0.6.

As contacts and dry friction between the grains plays a primary role in the dynamics of complex materials, special attention should be given to the numerical treatment of those phenomena. Different classes of approaches have been proposed in the literature, of which we can cite three:

- *Molecular Dynamics* (MD), which relax the assumption that distinct bodies cannot overlap, and use nonlinear springs to model the contacts between the particles (Cundall and Strack 1979). While this approach is the simplest to implement, limiting interpenetration can require the use of very stiff springs, which introduces a time scale much smaller than that of the macroscopic dynamics. This makes stable numerical integration difficult to achieve unless very small time steps are used, and may lead to visually disturbing flickering effects. Moreover, how to handle multi-valued friction laws in the MD framework is not obvious, and as such creeping residual motion may plague this approach.
- *Constraint-based approaches*, such as the *Non-Smooth Contact Dynamics* (NSCD; Jean 1999), propose to solve for each object's dynamics while ensuring that the Signorini-Coulomb conditions (1) and (3) are satisfied. While being inherently more complex than MD, constraint-based approaches allow the use of larger timesteps, and thus still prove computationally efficient. Chapter 2 will be dedicated to this kind of approaches, with an emphasis on the timestepping scheme proposed by Jean and Moreau (1987).

Scaling up The first part of this dissertation is dedicated to the simulation of dry frictional contacts in the DEM framework, using a constraint-based method; Chapter 4 presents results from the application of this strategy to hair dynamics. While those results were relatively good-looking, computational performance prevented us from simulating anything close to the whole $150,000$ individual fibers of a human head of hair. This motivated us to look for alternative approaches. Moreover, even though we were only simulating a small subset of the whole hair, we kept using a *fiber* model for each simulated strand, while a *wisp* model, representing the averaged behavior of several fibers, would have been more appropriate. Assuming the existence of such a model, we could as well take this approach one step further, and simulate the whole material using continuum mechanics.

As a first step towards the simulation of very large complex materials, the second part of this dissertation embraces this strategy, but focuses on simpler systems: granular materials consisting only of rigid grains. Devising a similar macroscopic simulation method for fibrous media such as hair remained out of the reach of this thesis.

[4] Note that the name DEM is also commonly used as a synonym for Molecular Dynamics; we use it here in a broader sense.

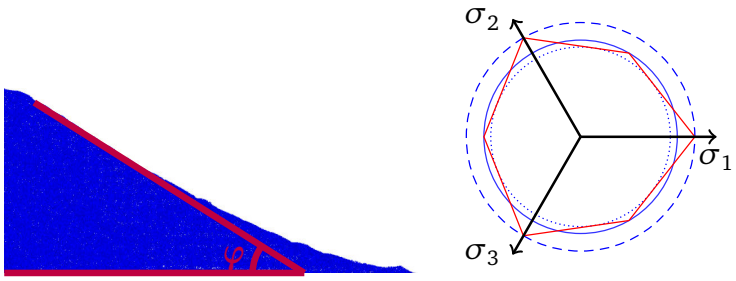


Figure 0.7: *Left: visualization of the friction angle φ on a 2D granular heap at rest. Right: 3D Mohr-Coulomb (red) with the three different Drucker-Prager yield surfaces (blue) in the plane of constant normal stress $\sum \boldsymbol{\sigma}_i = 3$.*

Using a continuum approach for granular materials makes sense, as they can be *extremely* large systems – a cubic meter of sand contains close to a trillion individual grains. We see immediately that simulating every grain as in the DEM framework, and taking into account each of its interactions with its neighbors, is not tractable, especially on standard computers. Moreover, the scale of inhomogeneities in granular materials is usually much smaller than the material itself. For these reasons, several constitutive laws have already been proposed in the literature to model the macroscopic behavior of granulars. The next section introduces fundamental concepts for the continuum modeling of dry frictional contact in granular materials.

0.3 Continuum modeling of dry friction

When the discrete constituents are sufficiently small w.r.t. the scale at which a phenomenon is studied for the material to appear spatially homogeneous, averaging processes may be used to intuit constitutive equations (or *rheologies*) on macroscopic quantities such as stress and strain.

0.3.1 Yield-stress flows

For instance, the presence of large molecules in so-called Bingham plastics such as mayonnaise manifest itself at the macroscopic scale by the onset of a *yield stress* σ_S. This means that irreversible deformations of the material will occur, i.e., the plastic strain rate $\dot{\boldsymbol{\epsilon}}$ will be non-zero, only once the norm of the deviatoric stress, $|\mathrm{Dev}\,\boldsymbol{\sigma}|$, has reached the critical value σ_S. Such materials may remain indefinitely stuck in various shapes, in contrast to Newtonian fluids which will always, albeit potentially slowly, go back to a flat shape. Note that different choices can be used for the definition of the norm $|\cdot|$ in the above expression, yielding slightly different rheologies, but an *objectivity* criterion should always be satisfied: one must ensure that the norm is invariant to changes in the reference frame. The Bingham model is classically defined using the second invariant of the deviatoric part of the stress tensor, $J_2(\boldsymbol{\sigma}) := \frac{1}{2}\mathrm{Tr}(\mathrm{Dev}\,\boldsymbol{\sigma})^2$, with $|\mathrm{Dev}\,\boldsymbol{\sigma}| := \sqrt{J_2(\boldsymbol{\sigma})}$. This definition ensures the objectivity of the model.

Such plastic phenomena are commonly described with a *yield surface*, that is, a function F, objective w.r.t. the stress tensor, such that the material remains solid while $F(\boldsymbol{\sigma}) < 0$, and $F(\boldsymbol{\sigma}) = 0$ corresponds to the flowing regime where irreversible deformation occurs. The yield surface for the Bingham model is given $F^{\mathrm{BI}}(\boldsymbol{\sigma}) := \sqrt{J_2(\boldsymbol{\sigma})} - \sigma_S$.

0.3.2 Frictional yield surfaces

Granular materials also exhibit a yield stress, as demonstrated by their ability to form heaps that do not (systematically) collapse over time. However, just like Coulomb friction featured a sliding threshold proportional to the normal applied load, the yield stress of dry granular materials is observed to depend on the normal (or *mean*) stress, i.e., the internal pressure.

Mohr–Coulomb criterion The Mohr–Coulomb (MC) criterion is the continuum mechanics generalization of Euler's inclined plane experiment, and consider the maximum angle φ (the so-called *rest angle*) that the slope of a granular heap can make without starting to avalanche (Figure 0.7, left). Let us first consider the 2D case, and let σ_φ and τ_φ denote the normal and shear stresses acting on an inclined plane of angle φ. By analogy with Euler's criterion, the stability of the granular heap up to an angle φ means that the material's yield condition can be written $\tau_\varphi \leq \sigma_\varphi \tan\varphi$. Mohr's circle relates σ_φ and τ_φ to the principal stresses of the material $\sigma_1 \leq \sigma_2$, i.e., the eigenvalues of the *applied* stress tensor, as

$$\sigma_\varphi = -\frac{\sigma_1+\sigma_2}{2} - \frac{\sigma_2-\sigma_1}{2}\sin\varphi, \qquad \tau_\varphi = \frac{\sigma_2-\sigma_1}{2}\cos\varphi.$$

The cohesionless MC criterion thus states that a granular material with rest angle φ will remain stable as long as

$$\begin{aligned}\left(-\frac{\sigma_1+\sigma_2}{2} - \frac{\sigma_2-\sigma_1}{2}\sin\varphi\right)\tan\varphi &\geq \frac{\sigma_2-\sigma_1}{2}\cos\varphi \\ -\frac{\sigma_1+\sigma_2}{2}\sin\varphi &\geq \frac{\sigma_2-\sigma_1}{2}. \end{aligned} \tag{4}$$

The 3D version of the Mohr-Coulomb criterion considers each plane of maximum shear, and can be summarized as

$$-\frac{\sigma_1+\sigma_3}{2}\sin\varphi \geq \frac{\sigma_3-\sigma_1}{2}, \tag{5}$$

where $\sigma_1 \leq \sigma_2 \leq \sigma_3$ are the eigenvalues of the material's stress. Inequation (5) written for each potential ordering of the eigenvalues defines 6 yield planes in principal stresses space; the MC yield surface is thus an hexagon-shaped convex cone centered around the hydrostatic axis $\sigma_1 = \sigma_2 = \sigma_3$, as illustrated in Figure 0.7, right. This hexagon degenerates to a triangle for $\sin\varphi = 1$, and approaches a regular (yet vanishing) hexagon for $\sin\varphi = 0$.

Note that while Coulomb friction is an approximation of interactions induced by the microscopic asperities of the contacting surface between grains, Mohr–Coulomb theory averages grain-sized inhomogeneities, and is thus only valid at a much bigger scale.

Drucker–Prager yield criterion The Mohr–Coulomb criterion (5) involves the individual eigenvalues of the stress tensor, and is numerically unwieldy. Drucker and Prager (1952) proposed a yield surface that is defined using only invariants of the stress tensor based on the Bingham model, but with a yield-stress that grows linearly with the first invariant of the stress tensor, $I_1(\boldsymbol{\sigma}) := \operatorname{Tr}\boldsymbol{\sigma}$. In the cohesionless case, the Drucker–Prager (DP) criterion is thus

$$\sqrt{J_2(\boldsymbol{\sigma})} + \hat{\mu}\frac{I_1(\boldsymbol{\sigma})}{d} \leq 0, \tag{6}$$

and $\hat{\mu}$ is called the friction coefficient.

Note that in $2D$, the Mohr–Coulomb and Drucker–Prager yield surfaces coincide. Indeed, $I_1(\boldsymbol{\sigma}) = \sigma_1+\sigma_2$, and $J_2(\boldsymbol{\sigma}) = \frac{1}{4}(\sigma_1-\sigma_2)^2$; Equations (6) and (4) thus become equivalent when $\hat{\mu} = \tan\varphi$.

However, in 3D, direct computations yield $J_2(\boldsymbol{\sigma}) = \frac{1}{6}\sum_{i\neq j}(\sigma_i-\sigma_j)^2$; the Drucker–Prager yield surface is thus a convex cone spanned by a circle centered on the hydrostatic axis (i.e., a Second-Order Cone). There is no hope for the DP and MC surfaces to fully match, but one may still choose the friction coefficient $\hat{\mu}$ using several heuristics (illustrated in Figure 0.7, right):

- $\hat{\mu} = \frac{2\sqrt{3}\sin\varphi}{3-\sin\varphi}$, so that DP circumscribes MC;
- $\hat{\mu} = \frac{\sin\varphi}{\sqrt{1+\frac{1}{3}\sin^2\varphi}}$, so that DP inscribes MC;
- $\hat{\mu} = \frac{2\sqrt{3}\sin\varphi}{3+\sin\varphi}$, so that DP interpolates MC at middle vertices.

Choice between these different values is application-dependent. For instance, risk-assessment simulations may want to use the inscribed surface, so that the predicted run-out length of an avalanche with DP will always overestimate the one using MC.

Cohesion and tensile strength The Drucker–Prager model can be extended to the modeling of cohesive materials, modifying the yield surface as (Alejano and Bobet 2012)

$$F^{\mathrm{DP}}_{\hat{\mu},\hat{c}}(\boldsymbol{\sigma}) := \hat{\mu}\frac{I_1(\boldsymbol{\sigma})}{d} - \hat{c} + \sqrt{J_2(\boldsymbol{\sigma})}. \tag{7}$$

A slightly more complex yield surface, the so-called Drucker–Prager yield surface with *tension cut-off*, may also be of interest for materials such as concrete. The cut-off dictates that the material will break when the mean tensile stress exceeds a critical value $\widehat{\tau_c}$,

$$F^{\mathrm{DP}}_{\hat{\mu},\hat{c},\widehat{\tau_c}}(\boldsymbol{\sigma}) := \max\left(I_1(\boldsymbol{\sigma}) - d\widehat{\tau_c}, F^{\mathrm{DP}}_{\hat{\mu},\hat{c}}(\boldsymbol{\sigma})\right). \tag{8}$$

The cut-off stress will influence the set of admissible stresses set only when there holds simultaneously $I_1(\boldsymbol{\sigma}) \geq d\widehat{\tau_c}$ and $\hat{\mu}I_1(\boldsymbol{\sigma}) \leq d\hat{c}$, which means $\hat{\mu}\widehat{\tau_c} \leq \hat{c}$; the original Drucker–Prager yield surface is retrieved when $\hat{\mu}\widehat{\tau_c} \geq \hat{c}$. For this reason, we will prefer parameterizing the yield surface (9) with a shear yield stress, $\sigma_S := \hat{c} - \hat{\mu}\widehat{\tau_c}$, rather than with the cohesion coefficient $\hat{c}$. In the following, we will thus write the Drucker–Prager yield surface with tension cut-off as

$$F^{\mathrm{DP}}_{\hat{\mu},\sigma_S,\widehat{\tau_c}}(\boldsymbol{\sigma}) := \max\left(I_1(\boldsymbol{\sigma}) - d\widehat{\tau_c}, \hat{\mu}\frac{I_1(\boldsymbol{\sigma}) - d\widehat{\tau_c}}{d} - \sigma_S + \sqrt{J_2(\boldsymbol{\sigma})}\right). \tag{9}$$

Note that the Bingham yield surface is recovered when $\widehat{\tau_c} = +\infty$.

Other yield surfaces Both the Mohr–Coulomb and Drucker–Prager yield surfaces are nonsmooth; the normal to the surface is not uniquely defined everywhere, in particular for $\boldsymbol{\sigma} = \mathbf{0}$ in the cohesionless case. As we will see in Chapter 1, this complicates the definition of a flow rule, that is, we will not be able to unambiguously express the direction of plastic displacement as a function of the stress tensor. ‘ Mast (2013) presents different strategies to circumvent this difficulty, such as using a smooth cap for the Drucker–Prager cone, or prescribing the flow to be along the hydrostatic axis when $\boldsymbol{\sigma} = \mathbf{0}$. Another interesting option that they explore is the use of the Matzuo–Nakai yield surface, which is smooth everywhere and better matches the hexagonal shape of the Mohr–Coulomb surface that the Drucker–Prager law.

0.3.3 Shearing granular flows

The “GDR MiDi” group (GDR MiDi 2004) studied dense granular shearing flows, and proposed a new constitutive law that was able to match experiments quantitatively, the so-called $\mu(I)$ rheology (Jop et al. 2006). Based on the Drucker–Prager yield criterion, this rheology suggests to vary the friction coefficient with the *inertial number* I,

$$I(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}) := \frac{\sqrt{J_2(\dot{\boldsymbol{\varepsilon}})}D_g}{\sqrt{I_1(\boldsymbol{\sigma})/(d\rho_g)}},$$

where $\dot{\boldsymbol{\varepsilon}}$ is the strain rate, D_g the average diameter of grains, and ρ_g their density. This dimensionless number relates the fluctuation of the velocity at the grain scale to that of the macroscopic flow. When $I = 0$, the material behaves like a solid, and for very high values of I, the material becomes akin to a gas; in between lies the dense flowing regime.

Shear-hardening friction The higher the inertial number, the more energy will be dissipated by grain–grain interactions, and the higher the friction coefficient should be. Jop et al. (2006) propose the following expression:

$$\mu(I) = \mu_S + \frac{\mu_D - \mu_S}{I_0/I + 1},$$

with $\mu_D \geq \mu_S$ and where I_0 is a material-dependent constant. In contrast to discrete friction laws which are usually taken to be slip-weakening (e.g., Coulomb friction when $\mu_S > \mu_D$), the $\mu(I)$ rheology is thus shear-hardening.

Note that the $\mu(I)$ rheology does not influence the rest angle of the material; at rest, $I = 0$ and $\mu(I) = \mu_S$.

Dilatancy In a similar manner, the volume fraction of grains ϕ, that is, the fraction of space occupied by the granular material, can be affected by the shear rate; when this translates into an augmentation of the flow volume, this phenomenon is known as *dilatancy*. In dense shearing flow, the volume fraction has been observed to decrease with the inertial number (GDR MiDi 2004). Roux and Radjai (1998) define the dilatancy angle ψ from the ratio between the volumetric and shear strain rates, $\frac{1}{d} I_1(\dot{\boldsymbol{\varepsilon}}) \tan\psi = J_2(\dot{\boldsymbol{\varepsilon}})$. They relate this angle to a critical volume fraction ϕ_c as $\psi(\phi) = \psi_0(\phi - \phi_c)$, so that positive dilatancy occurs above the critical volume fraction, but shear tends to compress the material when $\phi < \phi_{\max}$.

0.3.4 Other complex materials

This section has made clear that devising macroscopic laws for granular materials, even when assuming perfectly rigid and spherical grains, was already complex. Continuum modeling of assemblies of elastic and anisotropic objects such as fibers would require much more sophisticated models, and was thus left out of the scope of this dissertation.

0.4 Synopsis

This dissertation will be divided in two parts. The first one will be mostly dedicated to Coulomb friction, going from the modeling to the numerical resolution of frictional contacts between discrete bodies. Note that the first three chapters will consist mostly in a walk through standard models and numerical methods from the literature, which Chapter 4 will present original contributions.

- Chapter 1 will first look at the mathematical structure of Signorini–Coulomb conditions, presenting a few useful reformulations and showing where such law can fit in standard plasticity theory. Structural similarities with Drucker–Prager flows will be made explicit.
- Chapter 2 will go through standard modeling of contacts in discrete mechanical systems. The Moreau–Jean scheme will be presented, and we will show that each timestep can be reduced to one or more instances of a canonical problem, which we will call a *Discrete Coulomb Friction Problem* (DCFP).
- Chapter 3 will be dedicated to numerical algorithms for solving the DCFP, discussing their relative relevance for particular problem structures.
- Chapter 4 will present an original variant of the Gauss–Seidel algorithm that proved to perform very well on Computer Graphics applications, such as hair dynamics, hair inverse modeling, and cloth dynamics.

The second part will be focused on the continuum simulation of granular materials. Taking advantage of the similarities between the Coulomb friction law and the Drucker–Prager yield surface, we will show how the numerical methods devised for discrete mechanics can still be relevant in the continuum limit.

- Chapter 5 will serve as an introduction to numerical methods in continuum mechanics, and the different strategies use for the simulation of granular materials.
- Chapter 6 will present our proposed numerical method for the simulation of dense, dry granular flows, which will relax the classical incompressibility assumption and take profit of the DCFP formalism.

- Chapter 7 will extend this method to more general flows with varying volume fraction of grains. A discretization scheme and numerical resolution strategy focused on improving relevance to Computer Graphics applications will also be presented.
- Finally, Chapter 8 will discuss the extension to granular flows in which the interactions with a surrounding fluid cannot be neglected. Once again, we will show that the dynamics of such flows can be numerically solved as a sequence of DCFP.

Software and Publications

The numerical methods presented throughout this dissertation have been accompanied by the development of associated simulation programs. Some of them have been released as open-source: a sparse block matrix linear algebra library featuring a few DCFP solvers[5] and a granular simulation software[6] based on the Material Point Method framework of Chapter 7. The approach proposed in the first part of this manuscript has also been implemented in industrial settings, both for cosmetology and visual effects applications.

Moreover, some of the contributions presented in this dissertation have already been published. These publications are listed below, together with pointers to the corresponding parts of this manuscript.

Peer-reviewed journals

G. Daviet and F. Bertails-Descoubes (2016a). "A Semi-Implicit Material Point Method for the Continuum Simulation of Granular Materials". In: *ACM Transactions on Graphics*. SIGGRAPH '16 Technical Papers 35.4, p. 13. DOI: 10.1145/2897824.2925877 (Chapter 7)

G. Daviet and F. Bertails-Descoubes (2016b). "Nonsmooth simulation of dense granular flows with pressure-dependent yield stress". In: *Journal of Non-Newtonian Fluid Mechanics* 234, pp. 15–35. ISSN: 0377-0257. DOI: http://dx.doi.org/10.1016/j.jnnfm.2016.04.006 (Chapter 6)

A. Derouet-Jourdan, F. Bertails-Descoubes, G. Daviet, et al. (2013). "Inverse dynamic hair modeling with frictional contact". In: *ACM Transactions on Graphics* 32.6, pp. 1–10. ISSN: 0730-0301. DOI: 10.1145/2508363.2508398 (Chapter 4, Section 4.4.1)

G. Daviet, F. Bertails-Descoubes, and L. Boissieux (2011). "A hybrid iterative solver for robustly capturing Coulomb friction in hair dynamics". In: *ACM Transactions on Graphics* 30.6, pp. 1–12. ISSN: 0730-0301. DOI: 10.1145/2070781.2024173 (Chapter 4, Sections 4.1 and 4.2)

F. Bertails-Descoubes et al. (2011). "A nonsmooth Newton solver for capturing exact Coulomb friction in fiber assemblies". In: *ACM Transactions on Graphics* 30 (1), 6:1–6:14. ISSN: 0730-0301. DOI: http://doi.acm.org/10.1145/1899404.1899410 (Chapter 3, Section 3.1.2)

Technical reports

R. Casati et al. (2016). *Inverse Elastic Cloth Design with Contact and Friction*. Research Report. Inria Grenoble Rhône-Alpes, Université de Grenoble. URL: https://hal.archives-ouvertes.fr/hal-01309617 (Chapter 4, Section 4.4.2)

O. Bonnefon and G. Daviet (2011). *Quartic formulation of Coulomb 3D frictional contact*. Anglais. Tech. rep. INRIA - Laboratoire Jean Kuntzmann. URL: http://hal.archives-ouvertes.fr/inria-00553859/en/ (Chapter 4, Section 4.1.2)

Poster

G. Daviet, F. Bertails-Descoubes, and R. Casati (2015). "Fast cloth simulation with implicit contact and exact coulomb friction". In: *Proceedings of the 14th ACM SIGGRAPH / Eurographics Symposium on Computer Animation - SCA '15*. DOI: 10.1145/2786784.2795139 (Chapter 4, Section 4.3)

[5] bogus: http://gdaviet.fr/code/bogus

[6] Sand6: http://bipop.inrialpes.fr/~gdaviet/code/sand6

Part I

Numerical treatment of friction in discrete contact mechanics

1 Mathematical structure of Coulomb friction

In this chapter, we present a few equivalent formulations of the Coulomb law that will prove useful for the construction of numerical algorithms. Most of the properties compiled below stem from the works of Pierre Alart and Géry de Saxcé in the early 1990's, and rely upon convex analysis tools that were largely developed by Jean Jacques Moreau and R. Tyrrell Rockafellar in the 1960's. We refer the reader to Appendix A for a brief introduction to this theory, and the enunciation of a few properties of convex cones, subdifferentials and convex conjugates that we will make frequently use of. Note that this chapter is quite heavy on notations, and not fully required to follow the remainder of this dissertation — the equivalences established here will be appropriately referred to in the following chapters. The casual reader may choose to read Sections 1.1 and 1.3.1–1.3.3, and skip the remaining derivations.

1.1 Coulomb's friction law

1.1.1 Second-Order Cone

A convenient tool for expressing the Coulomb law and Drucker–Prager yield surface in an unified manner is the formalism of the Second-Order Cone, sometimes called Lorentz cone or ice-cream cone, which is a special case of convex cone (Definition A.10).

Definition 1.1 (Second-Order Cone)**.** *Let X be an Hilbert space, $\langle\cdot,\cdot\rangle$ a scalar product with associated norm $\|\cdot\| := \sqrt{\langle\cdot,\cdot\rangle}$, and $\boldsymbol{n} \neq 0 \in X$.*

For $\boldsymbol{x} \in X$, we note $\boldsymbol{x}_N \in \mathbb{R}$ the projection of $\boldsymbol{x}$ on the subspace spanned by $\boldsymbol{n}$, and $\boldsymbol{x}_T \in X$ the projection on its orthogonal complement, i.e., such that $\boldsymbol{x} = \boldsymbol{x}_N \frac{\boldsymbol{n}}{\|\boldsymbol{n}\|} + \boldsymbol{x}_T$ with $\langle \boldsymbol{x}_T, \boldsymbol{n}\rangle = 0$.

The Second-Order Cone of aperture $\mu \in \mathbb{R}_+ \cup \{+\infty\}$ w.r.t. $\boldsymbol{n}$ and $\langle\cdot,\cdot\rangle$ is the closed convex cone

$$\mathscr{K}_\mu(\boldsymbol{n}) := \left\{\boldsymbol{x} \in \mathscr{K}_\mu, \|\boldsymbol{x}_T\| \leq \mu \boldsymbol{x}_N\right\}.$$

Proof. The triangular inequality ensures that $\mathscr{K}_\mu(\boldsymbol{n})$ is indeed a convex cone. □

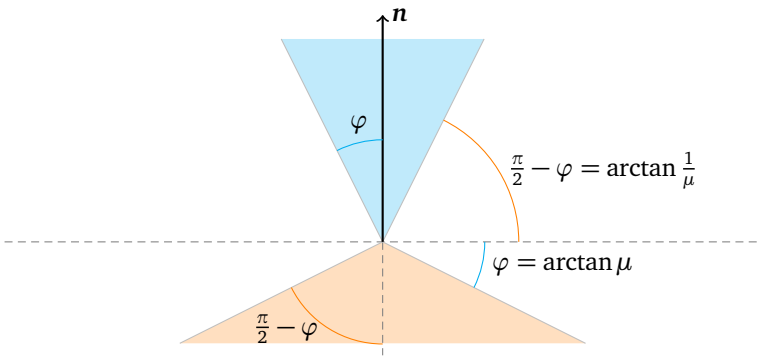
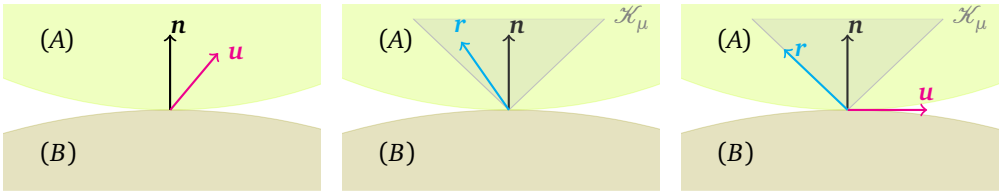


Figure 1.1: *The 2D Second-Order Cone of aperture μ, $\mathscr{K}_\mu(\boldsymbol{n})$ (above, cyan), and its polar cone $\mathscr{K}_{\frac{1}{\mu}}(-\boldsymbol{n})$ (below, orange)*



(a) *Take-off* **(b)** *Sticking* **(c)** *Sliding*

Figure 1.2: *The three cases of the disjunctive Coulomb frictional contact law*

An interesting property of the class of Second-Order Cones is that it is closed w.r.t. duality, that is, the dual cone to a SOC is also a SOC. Indeed, direct geometric computations illustrated in Figure 1.1 yield that:

Property 1.1. *The dual cone to the Second-Order Cone $\mathcal{K}_\mu(\boldsymbol{n})$ is $\mathcal{K}_{\frac{1}{\mu}}(\boldsymbol{n})$, the Second Order Cone of aperture $\frac{1}{\mu}$.*

1.1.2 Disjunctive formulation of the Signorini-Coulomb conditions

The Signorini (1) and Coulomb friction (3) conditions can be combined into a slightly more compact set of three cases, illustrated in Figure 1.2, which we will refer to as the *disjunctive* formulation of the Coulomb contact law (Cadoux 2009). In the *take-off* case, the two objects are separating and the reaction force vanishes; the *sticking* case holds when there is no relative motion; finally, the *sliding* corresponds to saturated friction and purely tangential relative motion. More formally, the relative velocity $\boldsymbol{u}$ and the contact force $\boldsymbol{r}$ should satisfy one of the following cases:

$$\left\{\begin{array}{lll} & \boldsymbol{r}=0 & \text{and } u_{\mathrm{N}}>0, \quad \text{(take-off)} \\ \text{or} & \boldsymbol{r}\in\mathcal{K}_\mu(\boldsymbol{n}) & \text{and } \boldsymbol{u}=0, \quad \text{(sticking)} \\ \text{or} & \left\{\begin{array}{l}\boldsymbol{r}\in \operatorname{Bd}\mathcal{K}_\mu(\boldsymbol{n}) \\ \boldsymbol{r}_{\mathbf{T}}=-\alpha\boldsymbol{u}_{\mathbf{T}},\ \alpha\in\mathbb{R}_+\end{array}\right. & \text{and } \left\{\begin{array}{l}u_{\mathrm{N}}=0 \\ \boldsymbol{u}_{\mathbf{T}}\neq\mathbf{0},\end{array}\right. \quad \text{(sliding)} \end{array}\right. \tag{1.1}$$

where $\mathcal{K}_\mu$ is defined w.r.t. the usual scalar product in $\mathbb{R}^d$. In the following, we will denote by $\mathscr{C}_\mu(\boldsymbol{n})\subset\mathbb{R}^d\times\mathbb{R}^d$ the set of velocity–force pairs satisfying the Signorini-Coulomb condition,

$$(\boldsymbol{u},\boldsymbol{r})\in\mathscr{C}_\mu(\boldsymbol{n}) \iff \boldsymbol{u} \text{ and } \boldsymbol{r} \text{ satisfy (1.1)} \iff \boldsymbol{u} \text{ and } \boldsymbol{r} \text{ satisfy (1) and (3)}.$$

For the sake of simplicity, we will also stop writing systematically the contact normal $\boldsymbol{n}$ relative to which the normal cone and the Coulomb law solution set are defined. That is, when the precise direction of the contact normal is of no relevance, we will write $\mathcal{K}_\mu$ instead of $\mathcal{K}_\mu(\boldsymbol{n})$ and $\mathscr{C}_\mu$ instead of $\mathscr{C}_\mu(\boldsymbol{n})$.

An analogous formulation may be derived by expressing the disjunction on $\boldsymbol{r}$ instead of $\boldsymbol{u}$,

$$(\boldsymbol{u},\boldsymbol{r})\in\mathscr{C}_\mu \iff \left\{\begin{array}{lll} & u_{\mathrm{N}}\geq 0 & \text{and } \boldsymbol{r}=\mathbf{0} \\ \text{or} & \boldsymbol{u}=\mathbf{0} & \text{and } \boldsymbol{r}\in\operatorname{int}\mathcal{K}_\mu \\ \text{or} & \left\{\begin{array}{l}u_{\mathrm{N}}=0 \\ \boldsymbol{u}_{\mathbf{T}}=-\alpha\boldsymbol{r}_{\mathbf{T}},\ \alpha\in\mathbb{R}_+\end{array}\right. & \text{and } \boldsymbol{r}\in\operatorname{Bd}\mathcal{K}_\mu\setminus\{\mathbf{0}\} \end{array}\right. \tag{1.2}$$

In Section 4.1.2 we will show that for a system with a single contact and a linear relationship between $\boldsymbol{u}$ and $\boldsymbol{r}$, we can find an analytical solution to the Signorini-Coulomb conditions by enumerating the cases of Equation (1.1). However, as soon as we have to deal with multiple contacts, the disjunctive formulations become cumbersome to work with. Indeed, for a system system with n contact points, one would have to check for the existence of a solution in each of the 3^n cases — the cost of such an enumeration would quickly become prohibitive. This

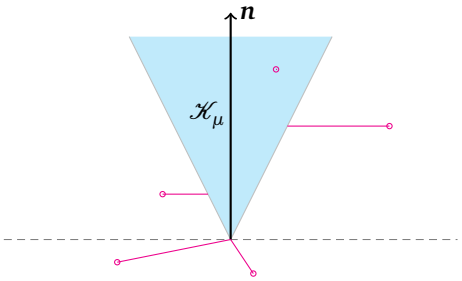


Figure 1.3: *The projection at the heart of the Alart–Curnier formulation*

motivates the search for alternative expressions of these conditions, i.e., ones that would lend themselves to numerical optimization methods.

1.1.3 Alart–Curnier function

Using the notion of normal cone (see Definition A.9), we can rewrite the Signorini and Coulomb friction conditions into a pair of root-finding problems.

Indeed, computing the normal cones of $\mathbb{R}_+$, the set of positive reals, and $\mathscr{B}^d(a) \subset \mathbb{R}^d$, the ball of radius $\max(0, a)$ centered at $\mathbf{0}$, yield

$$\mathscr{N}_{\mathbb{R}_+}(x) = \begin{cases} \emptyset & \text{if } x < 0 \\ \{0\} & \text{if } x > 0 \\ \mathbb{R}_- & \text{if } x = 0, \end{cases} \tag{1.3}$$

and

$$\mathscr{N}_{\mathscr{B}^d(a)}(\boldsymbol{x}) = \begin{cases} \emptyset & \text{if } \|\boldsymbol{x}\| > a \\ \{\mathbf{0}\} & \text{if } \|\boldsymbol{x}\| < a \\ \{\alpha \boldsymbol{x},\ \alpha \in \mathbb{R}_+\} & \text{if } \|\boldsymbol{x}\| = a \text{ and } a > 0 \\ \mathbb{R}^d & \text{if } \|\boldsymbol{x}\| = a \text{ and } a = 0. \end{cases} \tag{1.4}$$

Studying each case of those expressions confirms that they correspond to those of the Signorini and Coulomb friction laws. We thus get the equivalences

$$\begin{aligned} &(1) \iff \boldsymbol{u}_{\mathrm{N}} \in -\mathscr{N}_{\mathbb{R}_+}(\boldsymbol{r}_{\mathrm{N}}), \\ &(3) \iff \boldsymbol{u}_{\mathrm{T}} \in -\mathscr{N}_{\mathscr{B}^{d-1}(\mu r_{\mathrm{N}})}(\boldsymbol{r}_{\mathrm{T}}). \end{aligned}$$

Finally, we can use Corollary A.6 to express the normal cone inclusions as fixed points of orthogonal projections, and obtain the functional formulation of the Coulomb contact law introduced by Alart and Curnier (1991),

$$(\boldsymbol{u}, \boldsymbol{r}) \in \mathscr{C}_\mu \iff f_{\mathrm{AC}}(\boldsymbol{u}, \boldsymbol{r}) = \mathbf{0} \tag{1.5}$$

with, for any $\xi \in \mathbb{R}_+^*$,

$$\begin{array}{rccl} f_{\mathrm{AC}}: & \mathbb{R}^d \times \mathbb{R}^d & \to & \mathbb{R}^d \\ & (\boldsymbol{u}, \boldsymbol{r}) & \mapsto & \begin{pmatrix} \Pi_{\mathbb{R}_+} & (\boldsymbol{r}_{\mathrm{N}} - \xi \boldsymbol{u}_{\mathrm{N}}) \\ \Pi_{\mathscr{B}^{d-1}(\mu r_{\mathrm{N}})} & (\boldsymbol{r}_{\mathrm{T}} - \xi \boldsymbol{u}_{\mathrm{T}}) \end{pmatrix} - \boldsymbol{r}. \end{array} \tag{1.6}$$

The effect of the tangential and normal projections in Equation 1.6 is illustrated in Figure 1.3; note that their result is always in $\mathscr{K}_\mu$. Now, given a kinematic relationship between $\boldsymbol{u}$ and $\boldsymbol{r}$, one could express the search of solutions satisfying the Coulomb law as a root-finding problem, or as a minimization problem on the norm of f_{AC}. Note however that the orthogonal projections are

not differentiable everywhere, making f_{AC} a *nonsmooth* function, and thus requiring a careful design of the minimization algorithm.

While being an effective tool for the resolution of practical problems, the Alart–Curnier function yields little insight into the mathematical structure of the Coulomb contact law. One may wonder if we could retrieve more information from the theory of plasticity, which is what we will discuss in the following section.

1.2 Implicit Standard Materials

1.2.1 Generalized Standard Materials

In its simplest form, the theory of Generalized Standard Materials (GSM) relates the evolution of a strain-like variable, $\boldsymbol{\varepsilon} \in X$, and a stress-like variable, $\boldsymbol{\sigma} \in Y$, where X and Y are reflexive Banach spaces, dual for the bilinear form $\langle \cdot, \cdot \rangle$. Considering an isothermal system described by $\boldsymbol{\varepsilon}$ and another internal variable $\boldsymbol{\gamma}$, its evolution should satisfy (see, e.g., Saramito 2015, chapter 5)

$$\begin{cases} \boldsymbol{\sigma} \in \rho \dfrac{\partial \mathscr{E}}{\partial \boldsymbol{\varepsilon}}(\boldsymbol{\varepsilon}, \boldsymbol{\gamma}) + \dfrac{\partial \mathscr{D}}{\partial \dot{\boldsymbol{\varepsilon}}}(\dot{\boldsymbol{\varepsilon}}, \dot{\boldsymbol{\gamma}}) \\ \mathbf{0} \in \rho \dfrac{\partial \mathscr{E}}{\partial \boldsymbol{\gamma}}(\boldsymbol{\varepsilon}, \boldsymbol{\gamma}) + \dfrac{\partial \mathscr{D}}{\partial \dot{\boldsymbol{\gamma}}}(\dot{\boldsymbol{\varepsilon}}, \dot{\boldsymbol{\gamma}}) \end{cases}$$

with $\mathscr{E}$ and $\mathscr{D}$ proper closed convex functions, ρ is the density of the material, and $\dot{\boldsymbol{\varepsilon}} := \frac{\mathrm{d}\boldsymbol{\varepsilon}}{\mathrm{d}t}$. $\mathscr{E}$ is called the Helmotz free-energy function, and $\mathscr{D}$ the dissipation potential.

Still following Saramito (2015, chapter 5), we can decompose the strain $\boldsymbol{\varepsilon}$ as an elastic part, $\boldsymbol{\varepsilon}_e$, and a plastic part, $\boldsymbol{\varepsilon}_p$. Assuming locally small deformations, we write $\dot{\boldsymbol{\varepsilon}} = \dot{\boldsymbol{\varepsilon}}_e + \dot{\boldsymbol{\varepsilon}}_p$. We furthermore consider that the free energy depends only on $\boldsymbol{\varepsilon}_e$, and the potential of dissipation only on $\boldsymbol{\varepsilon}_p$. Writing again our equations for $\boldsymbol{\gamma} := \boldsymbol{\varepsilon}_e$, we get

$$\begin{cases} \boldsymbol{\sigma} \in \dfrac{\partial \mathscr{D}}{\partial \dot{\boldsymbol{\varepsilon}} - \dot{\boldsymbol{\varepsilon}}_e}(\dot{\boldsymbol{\varepsilon}} - \dot{\boldsymbol{\varepsilon}}_e) \\ 0 \in \rho \dfrac{\partial \mathscr{E}}{\partial \boldsymbol{\varepsilon}_e}(\boldsymbol{\varepsilon}_e) - \dfrac{\partial \mathscr{D}}{\partial \dot{\boldsymbol{\varepsilon}} - \dot{\boldsymbol{\varepsilon}}_e}(\dot{\boldsymbol{\varepsilon}} - \dot{\boldsymbol{\varepsilon}}_e), \end{cases}$$

or equivalently,

$$\begin{cases} \boldsymbol{\sigma} \in \dfrac{\partial \mathscr{D}}{\partial \dot{\boldsymbol{\varepsilon}}_p}(\dot{\boldsymbol{\varepsilon}}_p) & (1.7) \\ \boldsymbol{\sigma} \in \rho \dfrac{\partial \mathscr{E}}{\partial \boldsymbol{\varepsilon}_e}(\boldsymbol{\varepsilon}_e). & (1.8) \end{cases}$$

Note the purely plastic case is recovered for $\mathscr{E} = \mathscr{I}_{\{\mathbf{0}\}}$, as Equation (1.8) then boils down to $\boldsymbol{\varepsilon}_e = \mathbf{0}$.

Using Theorem A.2, we can also write Equation (1.7) as the so-called *flow rule*,

$$\dot{\boldsymbol{\varepsilon}}_p \in \frac{\partial \mathscr{D}^\star}{\partial \boldsymbol{\sigma}}(\boldsymbol{\sigma}). \tag{1.9}$$

We are interested in the particular case of yield-stress materials, where analogously to our Coulomb contact force $\boldsymbol{r}$ that must lie in $\mathscr{K}_\mu$, the stress $\boldsymbol{\sigma}$ must belong to an admissible set. Let $C := \{\boldsymbol{\sigma},\ F(\boldsymbol{\sigma}) \leq 0\}$ be this set of admissible stresses, with $F(\boldsymbol{\sigma}) = 0$ the yield isosurface. The flow rule (1.9) is said to be associated if it is derived from Hill's maximum power principle (see e.g., de Saxcé and Bousshine 2002), which imposes

$$\boldsymbol{\sigma} = \operatorname*{arg\,min}_{\tau \in C} -\left\langle \boldsymbol{\tau}, \dot{\boldsymbol{\varepsilon}}_p \right\rangle.$$

From the optimality condition of Theorem A.6, a flow rule will thus be associated if and only if $\dot{\boldsymbol{\varepsilon}}_p \in \mathscr{N}_C(\boldsymbol{\sigma})$, and therefore $\mathscr{D} = \mathscr{I}_C{}^\star$.

Remark The flow rule (1.9) is often written as

$$\dot{\boldsymbol{\varepsilon}}_p \in \frac{\partial \mathscr{D}^\star}{\partial \boldsymbol{\sigma}}(\boldsymbol{\sigma}) \iff \dot{\boldsymbol{\varepsilon}}_p \in \alpha \frac{\partial F}{\partial \boldsymbol{\sigma}}(\boldsymbol{\sigma}) \text{ and } \begin{cases} F(\boldsymbol{\sigma}) \leq 0 & \\ \quad \alpha = \mathbf{0} & \text{if } F(\boldsymbol{\sigma}) < 0 \\ \quad \alpha \in \mathbb{R}_+ & \text{if } F(\boldsymbol{\sigma}) = 0 \end{cases}$$

$$\iff \dot{\boldsymbol{\varepsilon}}_p \in \alpha \frac{\partial F}{\partial \boldsymbol{\sigma}}(\boldsymbol{\sigma}) \text{ and } 0 \geq F(\boldsymbol{\sigma}) \perp \alpha \geq 0$$

where α is called the plastic multiplier, or consistency parameter. The equivalence – under regularity conditions for F – is granted by Property (A.16).

Coulomb law In our multibody contact setting, $-\boldsymbol{r}$ plays the role of $\boldsymbol{\sigma}$ and $\boldsymbol{u}$ that of $\dot{\boldsymbol{\varepsilon}}_p$. As a first attempt to fit Coulomb friction in the GSM framework, we can define an associated flow rule from the admissible set for the contact force, $-\boldsymbol{r} \in -\mathscr{K}_\mu$. Since $\mathscr{K}_\mu$ is a convex cone, Theorem A.4 states that the flow rule $\boldsymbol{u} \in \mathscr{N}_{-\mathscr{K}_\mu}(-\boldsymbol{r})$ will be equivalent to $\mathscr{K}_\mu \ni \boldsymbol{r} \perp \boldsymbol{u} \in \mathscr{K}_{\frac{1}{\mu}}$. This means that the power dissipated through the friction force, $\langle \boldsymbol{u}, \boldsymbol{r} \rangle$, will always be zero; this is physically absurd, unless $\mu = 0$. We therefore conclude that Coulomb friction cannot be modeled with an associated flow rule. Actually, de Saxcé and Feng (1998) showed that attempting to fit the Coulomb contact law inside the GSM framework is vain, and that Rockafellar's cyclic monotonicity criterion asserts that one cannot find a dissipation potential $\mathscr{D}$ such that $(\boldsymbol{u}, \boldsymbol{r}) \in \mathscr{C}_\mu \iff -\boldsymbol{r} \in \partial \mathscr{D}(\boldsymbol{u})$. Instead, they introduce the Implicit Standard Material framework, a superset of GSM in which the dissipation potential may depend on the strain-rate-like variables.

1.2.2 Implicit Standard Materials

The theory of Implicit Standard Materials (ISM) was introduced by de Saxcé and Feng (1991), and extended in (Berga and de Saxcé 1994; de Saxcé 1992; de Saxcé and Feng 1998). As the standard notion of dissipation potential is not adapted to the modeling of Coulomb's law, they introduce the weaker notion of *bipotentials*, which they define as follow:

Definition 1.2 (Bipotential)**.** *A function $b : X \times Y \to \bar{\mathbb{R}}$ is called a bipotential if*

1. *b is convex and closed w.r.t. each of its two parameters, and*
2. *b satisfies a Fenchel-Young-like inequality, that is*

$$b(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}) \geq \langle \dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma} \rangle \quad \forall \dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma} \in X \times Y. \tag{1.10}$$

This definition allows for a weaker, unilateral version of Theorem A.2,

Property 1.2. *If b is a bipotential, then*

$$b(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}) = \langle \dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma} \rangle \implies \dot{\boldsymbol{\varepsilon}} \in \frac{\partial b}{\partial \boldsymbol{\sigma}}(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}) \text{ and } \boldsymbol{\sigma} \in \frac{\partial b}{\partial \dot{\boldsymbol{\varepsilon}}}(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}).$$

Proof. Combining the left-hand-side with the inequality (1.10) yields

$$\begin{aligned} b(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\tau}) - b(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}) &\geq \langle \dot{\boldsymbol{\varepsilon}}, \boldsymbol{\tau} \rangle - \langle \dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma} \rangle && \forall \boldsymbol{\tau} \in Y, \text{ i.e.,} \\ b(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\tau}) &\geq b(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}) + \langle \dot{\boldsymbol{\varepsilon}}, \boldsymbol{\tau} - \boldsymbol{\sigma} \rangle && \forall \boldsymbol{\tau} \in Y, \end{aligned}$$

which by definition of the subdifferential means $\dot{\boldsymbol{\varepsilon}} \in \frac{\partial b}{\partial \boldsymbol{\sigma}}(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma})$. The second inclusion, $\boldsymbol{\sigma} \in \frac{\partial b}{\partial \dot{\boldsymbol{\varepsilon}}}(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma})$, can be derived in a similar manner. □

This motivates the Implicit Standard Material approach, which replaces Equations (1.7) and (1.9) from GSM with

$$\boldsymbol{\sigma} \in \frac{\partial b}{\partial \dot{\boldsymbol{\varepsilon}}_p}(\dot{\boldsymbol{\varepsilon}}_p, \boldsymbol{\sigma}) \tag{1.11}$$

$$\dot{\boldsymbol{\varepsilon}}_p \in \frac{\partial b}{\partial \boldsymbol{\sigma}}(\dot{\boldsymbol{\varepsilon}}_p, \boldsymbol{\sigma}). \tag{1.12}$$

One can readily see that for $b(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}) = \mathscr{D}(\dot{\boldsymbol{\varepsilon}}) + \mathscr{D}^\star(\boldsymbol{\sigma})$, we fall back to the GSM case and the reciprocal of Property 1.2 is also true. In the general case, the reverse implication will have to be checked manually to ensure that Equations (1.11) and (1.12) are equivalent.

Coulomb law In the framework of ISM, de Saxcé and Feng (1991) constructed a bipotential $b(\boldsymbol{u}, -\boldsymbol{r})$ such that Equations (1.11) and (1.12) are equivalent to the Coulomb contact law (1.1).

In order to heuristically construct such a bipotential, we can make the following considerations:

- The force $\boldsymbol{r}$ should lie in $\mathscr{K}_\mu$
- The relative velocity $\boldsymbol{u}$ should lie in $\mathscr{K}_\infty = \mathbb{R}_+ \times \mathbb{R}^{d-1}$
- The energy dissipation should be consistent with Coulomb's law, meaning

$$b(\boldsymbol{u}, -\boldsymbol{r}) = -\langle \boldsymbol{u}, \boldsymbol{r} \rangle = -\langle \boldsymbol{u}_{\mathrm{T}}, \boldsymbol{n}_{\mathrm{T}} \rangle = \mu \boldsymbol{r}_{\mathrm{N}} \|\boldsymbol{u}_{\mathrm{T}}\| \tag{1.13}$$

The simplest function satisfying those rules is indeed the one proposed by de Saxcé and Feng (1991),

$$b(\boldsymbol{u}, -\boldsymbol{r}) = \mathscr{I}_{-\mathscr{K}_\mu}(-\boldsymbol{r}) + \mathscr{I}_{\mathscr{K}_\infty}(\boldsymbol{u}) + \mu \|\boldsymbol{u}_{\mathrm{T}}\| \boldsymbol{r}_{\mathrm{N}}. \tag{1.14}$$

Property 1.3. *b defined as in (1.14) is a bipotential.*

Proof. The closed, convex nature of b w.r.t. each variable is immediate as b is the sum of such functions. Now, for $\boldsymbol{u} \in \mathscr{K}_\infty$ and $\boldsymbol{r} \in \mathscr{K}_\mu$, we have to show that

$$\begin{aligned} &b(\boldsymbol{u}, -\boldsymbol{r}) \geq -\langle \boldsymbol{u}, \boldsymbol{r} \rangle && \text{i.e.,} \\ &\mu \|\boldsymbol{u}_{\mathrm{T}}\| \boldsymbol{r}_{\mathrm{N}} \geq -\boldsymbol{u}_{\mathrm{N}} \boldsymbol{r}_{\mathrm{N}} - \langle \boldsymbol{u}_{\mathrm{T}}, \boldsymbol{r}_{\mathrm{T}} \rangle . \end{aligned}$$

Using the Cauchy-Schwartz inequality and the fact that $\boldsymbol{r} \in \mathscr{K}_\mu$,

$$-\langle \boldsymbol{u}_{\mathrm{T}}, \boldsymbol{r}_{\mathrm{T}} \rangle \leq \|\boldsymbol{u}_{\mathrm{T}}\| \|\boldsymbol{r}_{\mathrm{T}}\| \leq \mu \boldsymbol{r}_{\mathrm{N}} \|\boldsymbol{u}_{\mathrm{T}}\|.$$

Finally, $\boldsymbol{u}_{\mathrm{N}} \geq 0$ and $\boldsymbol{r}_{\mathrm{N}} \geq 0$, yielding $0 \geq -\boldsymbol{u}_{\mathrm{N}} \boldsymbol{r}_{\mathrm{N}}$ and thus concluding the proof. □

De Saxcé change of variable Now, let us show that with b defined as per (1.14), Equations (1.11) and (1.12) are equivalent to $(\boldsymbol{u}, \boldsymbol{r}) \in \mathscr{C}_\mu$. Still following de Saxcé and Feng, we first state an intermediate result; let us introduce the auxiliary variable $\tilde{\boldsymbol{u}}$, deduced from $\boldsymbol{u}$ as $\tilde{\boldsymbol{u}} := \boldsymbol{u} + \mu \|\boldsymbol{u}_{\mathrm{T}}\| \boldsymbol{n}$.

This change of variable $\boldsymbol{u} \mapsto \tilde{\boldsymbol{u}}$, illustrated in Figure 1.4 and which we will refer to as the *de Saxcé change of variable*, maps the half-space $\mathscr{K}_\infty$ to the Second-Order Cone $\mathscr{K}_{\frac{1}{\mu}}$, with

$$\begin{cases} \boldsymbol{u}_{\mathrm{N}} \geq 0 \iff \boldsymbol{u} + \mu \|\boldsymbol{u}_{\mathrm{T}}\| \boldsymbol{n} \in \mathscr{K}_{\frac{1}{\mu}} & \text{and} \\ \boldsymbol{u}_{\mathrm{N}} = 0 \iff \boldsymbol{u} + \mu \|\boldsymbol{u}_{\mathrm{T}}\| \boldsymbol{n} \in \operatorname{Bd} \mathscr{K}_{\frac{1}{\mu}}. \end{cases}$$

We get the following equivalences, which will prove very useful for the conception of numerical algorithms.

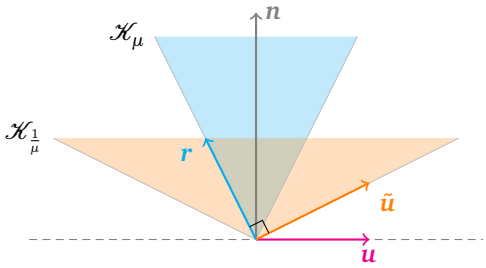


Figure 1.4: *De Saxcé change of variable $u \mapsto \tilde{u}$ in the sliding case of Coulomb friction. More generally, this change of variable maps the half-space $\mathscr{K}_\infty$ to the dual cone $\mathscr{K}_{\frac{1}{\mu}}$*

Property 1.4. *The Coulomb contact law (1.1) can be expressed in a compact manner on the variables r and $\tilde{u}$ as*

$$(u,r) \in \mathscr{C}_\mu \iff \tilde{u} \in -\mathscr{N}_{\mathscr{K}_\mu} r \iff r \in -\mathscr{N}_{\mathscr{K}_{\frac{1}{\mu}}} \tilde{u} \iff \mathscr{K}_{\frac{1}{\mu}} \ni \tilde{u} \perp r \in \mathscr{K}_\mu,$$

where the $\perp$ notation is referring to the usual $\mathbb{R}^d$ orthogonality.

Proof. Let us prove the first equivalence. Property A.14 gives us the expression of $\mathscr{N}_{\mathscr{K}_\mu}$,

$$\tilde{u} \in -\mathscr{N}_{\mathscr{K}_\mu}(r) \iff \begin{cases} r \in \mathscr{K}_\mu & \\ \tilde{u} = 0 & \text{if } r \in \operatorname{int} \mathscr{K}_\mu \\ \tilde{u} \in \mathscr{K}_{\frac{1}{\mu}} & \text{if } r = 0 \\ \tilde{u} \in \operatorname{Bd} \mathscr{K}_{\frac{1}{\mu}} \cap \{r\}^\perp & \text{if } r \in \operatorname{Bd} \mathscr{K}_\mu \setminus \{0\} \end{cases}$$

$$\iff \begin{cases} r \in \mathscr{K}_\mu & \\ u = 0 & \text{if } r \in \operatorname{int} \mathscr{K}_\mu \\ u \in \mathscr{K}_\infty & \text{if } r = 0 \\ u \in \operatorname{Bd} \mathscr{K}_\infty \text{ and } u_{\mathrm{T}} = -\alpha r_{\mathrm{T}},\ \alpha \in \mathbb{R}_+ & \text{if } r \in \operatorname{Bd} \mathscr{K}_\mu \setminus \{0\} \end{cases}$$

We recognize the disjunctive formulation of the Coulomb law on r, Equation (1.2), and conclude that $\tilde{u} \in -\mathscr{N}_{\mathscr{K}_\mu}(r) \iff (u,r) \in \mathscr{C}_\mu$. Then, the other equivalences are direct applications of Theorem A.4, noting that $-\mathscr{K}_\mu^\circ = \mathscr{K}_{\frac{1}{\mu}}$. □

We therefore want our candidate bipotential to satisfy $\tilde{u} \in \partial \mathscr{I}_{-\mathscr{K}_\mu}(-r) \iff u \in -\frac{\partial b}{\partial r}(u,-r)$. This will be the case if we choose b as $b(u,-r) = \mathscr{I}_{-\mathscr{K}_\mu}(-r) + \mu \|u_{\mathrm{T}}\| r_{\mathrm{N}} + f(u)$.

Property 1.5 (Bipotential for Coulomb law)**.** *The bipotential b defined as in Equation (1.14) models the Coulomb contact law, i.e.,*

$$b(u,-r) = -\langle u, r \rangle \iff r = -\frac{\partial b}{\partial u}(u,-r) \iff u = -\frac{\partial b}{\partial r}(u,-r) \iff (u,r) \in \mathscr{C}_\mu$$

Proof. From Properties 1.2 and 1.3, we know that

$$b(u,-r) = -\langle u, r \rangle \implies r = -\frac{\partial b}{\partial u}(u,-r) \text{ and } u = -\frac{\partial b}{\partial r}(u,-r).$$

As the regularity conditions of Corollary A.1 on the subdifferential of a sum are satisfied, we get

$$\begin{aligned} u \in -\frac{\partial b}{\partial r}(u,-r) &\iff u \in \left(\partial \mathscr{I}_{-\mathscr{K}_\mu}(-r) - \{\mu \|u_{\mathrm{T}}\| n\}\right) \\ &\iff \tilde{u} \in -\mathscr{N}_{-\mathscr{K}_\mu} && \text{using Property A.9} \\ &\iff (u,r) \in \mathscr{C}_\mu. && \text{using Property 1.4} \end{aligned}$$

Now, suppose $\boldsymbol{r} \in -\frac{\partial b}{\partial \boldsymbol{u}}(\boldsymbol{u}, -\boldsymbol{r})$, and let us show $b(\boldsymbol{u}, -\boldsymbol{r}) = -\langle \boldsymbol{u}, \boldsymbol{r} \rangle$. Using Theorem A.1 on the subdifferential of a sum and Property A.14 on the normal cone to a convex cone, we have $\boldsymbol{r} \in \mathscr{K}_0 \cap \{\boldsymbol{u}\}^\perp - \mu \boldsymbol{r}_{\mathrm{N}} \partial \|\boldsymbol{u}\| \boldsymbol{u}$. We always get $\boldsymbol{r} \in \mathscr{K}_\mu$, and can distinguish three cases:

1. $\boldsymbol{u} = \boldsymbol{0}$, then $b(\boldsymbol{u}, -\boldsymbol{r}) = 0 = -\langle \boldsymbol{u}, \boldsymbol{r} \rangle$.
2. $\boldsymbol{u}_{\mathrm{N}} \neq 0$ and $\boldsymbol{u}_{\mathbf{T}} = \boldsymbol{0}$, then once again $-\langle \boldsymbol{u}, \boldsymbol{r} \rangle = -\boldsymbol{u}_{\mathrm{N}} \boldsymbol{r}_{\mathrm{N}} = 0 = b(\boldsymbol{u}, -\boldsymbol{r})$.
3. $\boldsymbol{u}_{\mathrm{N}} \neq 0$ and $\boldsymbol{u}_{\mathbf{T}} \neq \boldsymbol{0}$, then $\boldsymbol{r} + \mu \boldsymbol{r}_{\mathrm{N}} \frac{\boldsymbol{u}_{\mathbf{T}}}{\|\boldsymbol{u}_{\mathbf{T}}\|} \in \{\boldsymbol{u}\}^\perp$, and therefore $-\langle \boldsymbol{u}, \boldsymbol{r} \rangle = \mu \boldsymbol{r}_{\mathrm{N}} \|\boldsymbol{u}_{\mathbf{T}}\| = b(\boldsymbol{u}, -\boldsymbol{r})$.

To conclude the proof, remark that the implication $(\boldsymbol{u}, \boldsymbol{r}) \in \mathscr{C}_\mu \implies b(\boldsymbol{u}, -\boldsymbol{r}) = -\langle \boldsymbol{u}, \boldsymbol{r} \rangle$ was ensured by our third criterion (1.13) in the heuristic construction of b. □

Remark We have seen that the Coulomb law is more complex than associated flow rules, in the sense that its bipotential includes a term that couples the strain and stress variables, $\boldsymbol{r}_{\mathrm{N}} \|\boldsymbol{u}_{\mathbf{T}}\|$; we can well imagine that this will lead to a tougher numerical problem. However, one can get back to the "easier" case of an associated law by temporarily freezing either one of the variables. Indeed we will see in Section 2.3.2 that freezing $\boldsymbol{r}_{\mathrm{N}}$ leads to the Haslinger (1983) algorithm (successive approximations with the Tresca law, yielding minimization problems over a half-cylinder), while freezing $\|\boldsymbol{u}_{\mathbf{T}}\|$ leads to the Cadoux (2009) algorithm (successive minimization problems over a Second-Order Cone).

1.3 Application to Drucker–Prager plasticity

The Drucker–Prager yield surface (7) is structurally very similar to the Coulomb friction law. Indeed, Moreau (1965) cites this application as a motivating factor for convex analysis, together with the study of unilateral contacts with friction. More recently, Berga and de Saxcé (1994) showed that the bipotential (1.14) could be easily adapted to model non-associated Drucker–Prager plasticity; we will keep their point of view below. Alternatively, we provided a less general but more direct derivation of the equivalence between Coulomb law and a particular case of the Drucker–Prager flow rule in (Daviet and Bertails-Descoubes 2016b, Section 3.2).

In order to derive such results, we first present a few notations to reason about the space to which the stress $\boldsymbol{\sigma}$ and displacements $\boldsymbol{\varepsilon}$ belong, i.e., the space of $d \times d$ symmetric tensors.

1.3.1 Symmetric tensors

Let us consider the space of $d \times d$ symmetric tensors S_d,

$$S_d = \left\{ (\boldsymbol{x} \otimes \boldsymbol{y} + \boldsymbol{y} \otimes \boldsymbol{x}),\ \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^{2d} \right\} \tag{1.15}$$

equipped with the scalar product $\langle \cdot, \cdot \rangle$ defined from the twice-contracted tensor product,

$$\langle \boldsymbol{\sigma}, \boldsymbol{\tau} \rangle := \frac{1}{2} \boldsymbol{\sigma} : \boldsymbol{\tau} := \frac{1}{2} \sum_{i,j} \boldsymbol{\tau}_{ij} \boldsymbol{\sigma}_{ij} \quad \forall \boldsymbol{\sigma}, \boldsymbol{\tau} \in S_d^2.$$

We note $|\cdot| := \sqrt{\langle \cdot, \cdot \rangle}$ the norm associated to this scalar product. Finally, we introduce the notation Tr for the linear form associating to each tensor its the trace, $\mathrm{Tr} : S_d \to \mathbb{R}$, $\boldsymbol{\tau} \mapsto \sum_i \boldsymbol{\tau}_{ii}$, and Dev for the linear application yielding its deviatoric part, $\mathrm{Dev} : S_d \to S_d$, $\boldsymbol{\tau} \mapsto \boldsymbol{\tau} - \frac{1}{d} \mathrm{Tr}\, \boldsymbol{\tau}$.

Using these notions, we can define a Second-Order Cone on S_d. First, notice that the element of S_d representing the linear form Tr through the scalar product $\langle \cdot, \cdot \rangle$ is $2\mathbb{I}_d$, i.e., $\mathrm{Tr}\, \boldsymbol{\tau} = \langle \boldsymbol{\tau}, 2\mathbb{I}_d \rangle$. We note $\boldsymbol{\iota}_d$ the corresponding unit tensor,

$$\boldsymbol{\iota}_d = \frac{2\mathbb{I}_d}{|2\mathbb{I}_d|} = \sqrt{\frac{2}{d}} \mathbb{I}_d.$$

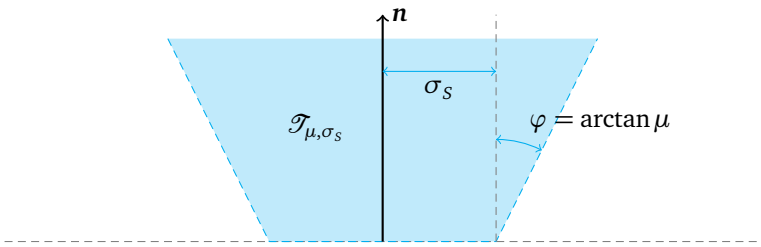


Figure 1.5: *The 2D truncated Second-Order Cone* $\mathscr{T}_{\mu,\sigma_S}$

Then, it can be readily seen that $\operatorname{Tr}(\operatorname{Dev}\boldsymbol{\tau}) = 0$, and that any symmetric tensor $\boldsymbol{\tau}$ can be decomposed as the orthogonal sum $\boldsymbol{\tau} = \langle \boldsymbol{\tau}, \boldsymbol{\iota}_d \rangle \boldsymbol{\iota}_d + \operatorname{Dev}\boldsymbol{\tau}$. We can therefore extend the notation of *normal part* and *tangential part* to the space of symmetric tensors with, for $\boldsymbol{\tau} \in S_d$,

$$\boldsymbol{\tau}_{\mathrm{N}} = \langle \boldsymbol{\tau}, \boldsymbol{\iota}_d \rangle \qquad\qquad \boldsymbol{\tau}_{\mathbf{T}} = \operatorname{Dev}\boldsymbol{\tau}.$$

In the following, we will reuse the notation $\mathscr{K}_\mu$ for the Second-Order Cone on the space of symmetric tensors defined w.r.t. $\langle \cdot, \cdot \rangle$ and the linear form Tr,

$$\mathscr{K}_\mu(\boldsymbol{\iota}_d) := \{\boldsymbol{\sigma} \in S_d, |\boldsymbol{\sigma}_{\mathbf{T}}| \leq \mu \boldsymbol{\sigma}_{\mathrm{N}}\} = \left\{\boldsymbol{\sigma} \in S_d, |\operatorname{Dev}\boldsymbol{\sigma}| \leq \frac{\mu}{\sqrt{2d}} \operatorname{Tr}\boldsymbol{\sigma}\right\}$$

1.3.2 Drucker–Prager yield surface

In order to study the Drucker–Prager yield surface in this framework, we have to express the invariants $I_1(\boldsymbol{\sigma})$ and $J_2(\boldsymbol{\sigma})$ as functions of the normal and tangential parts of the stress tensor $\boldsymbol{\sigma}$. We have directly $I_1(\boldsymbol{\sigma}) = \operatorname{Tr}\boldsymbol{\sigma} = \sqrt{2d}\boldsymbol{\sigma}_{\mathrm{N}}$, and

$$\begin{aligned} J_2(\boldsymbol{\sigma}) &= \frac{1}{2}\operatorname{Tr}\left[(\operatorname{Dev}\boldsymbol{\sigma})^2\right] = \frac{1}{2}\sum_i \left(\sum_j \left[(\operatorname{Dev}\boldsymbol{\sigma})_{ij}\right]^2\right) = |\operatorname{Dev}\boldsymbol{\sigma}|^2 \\ &= |\boldsymbol{\sigma}_{\mathbf{T}}|^2 \end{aligned}$$

The Drucker–Prager yield surface with tension cut-off is defined by Equation (9),

$$\begin{aligned} F^{\mathrm{DP}}_{\hat{\mu},\sigma_S,\widehat{\tau_c}}(\boldsymbol{\sigma}) &= \max\left(I_1(\boldsymbol{\sigma}) - d\widehat{\tau_c}, \hat{\mu}\frac{I_1(\boldsymbol{\sigma}) - d\widehat{\tau_c}}{d} - \sigma_S + \sqrt{J_2(\boldsymbol{\sigma})}\right) \\ &= \max\left(\sqrt{2d}\,(\boldsymbol{\sigma}_{\mathrm{N}} - \tau_c), \mu(\boldsymbol{\sigma}_{\mathrm{N}} - \tau_c) - \sigma_S + |\boldsymbol{\sigma}_{\mathbf{T}}|\right) \end{aligned}$$

using the notation $\mu := \sqrt{\frac{2}{d}}\hat{\mu}$ and $\tau_c = \sqrt{\frac{d}{2}}\widehat{\tau_c}$. We have thus the equivalence

$$F^{\mathrm{DP}}_{\hat{\mu},\sigma_S,\widehat{\tau_c}}(\boldsymbol{\sigma}) \leq 0 \iff \begin{cases} \tau_c - \boldsymbol{\sigma}_{\mathrm{N}} \geq 0 \\ \mu(\tau_c - \boldsymbol{\sigma}_{\mathrm{N}}) + \sigma_S \geq \|\boldsymbol{\sigma}_{\mathbf{T}}\|. \end{cases}$$

In the case $\sigma_S = 0$, we retrieve a set of of admissible stresses similar in structure to that of the Coulomb friction, a translation of the SOC $\mathscr{K}_\mu$:

$$F^{\mathrm{DP}}_{\hat{\mu},\hat{c}}(\boldsymbol{\sigma}) \leq 0 \iff (\tau_c \boldsymbol{\iota}_d - \boldsymbol{\sigma}) \in \mathscr{K}_\mu.$$

In the general case, for $\sigma_S \geq 0$, the set of admissible stresses is instead a translation of a *truncated* Second-Order Cone $\mathscr{T}_{\mu,\sigma_S}$,

$$\mathscr{T}_{\mu,\sigma_S} := \{\boldsymbol{\sigma} \in S_d, \boldsymbol{\sigma}_{\mathrm{N}} \geq 0 \text{ and } |\boldsymbol{\sigma}_{\mathbf{T}}| \leq \tau_c + \mu\boldsymbol{\sigma}_{\mathrm{N}}\}, \tag{1.16}$$

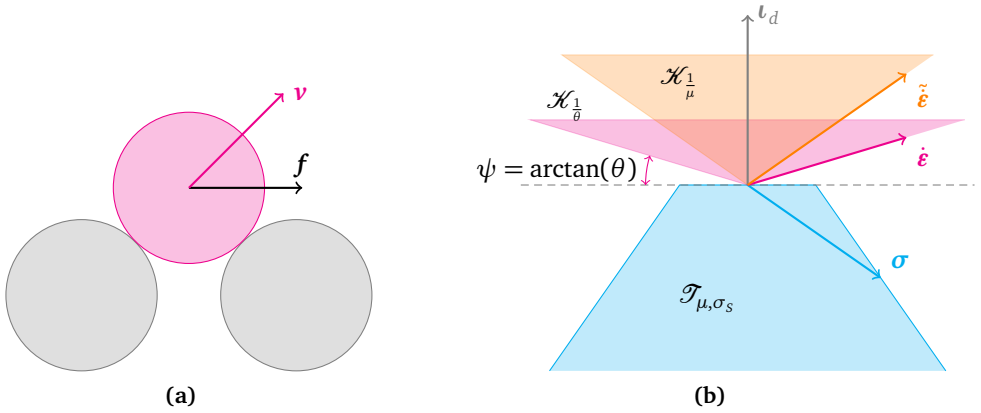


Figure 1.6: ***(a)*** *Rationale for dilatancy: the application of a shearing force* $\boldsymbol{f}$ *will make the "trapped" pink particle move upwards as well as horizontally.* ***(b)*** *De Saxcé change of variable in the flowing regime of the Drucker–Prager flow rule with dilatancy angle* ψ

depicted in Figure 1.5. We have the equivalence

$$F^{\mathrm{DP}}_{\hat{\mu},\sigma_S,\widehat{\tau_c}}(\boldsymbol{\sigma}) \leq 0 \iff (\tau_c \boldsymbol{\iota}_d - \boldsymbol{\sigma}) \in \mathscr{T}_{\mu,\sigma_S}. \tag{1.17}$$

Notice that another expression for $\mathscr{T}_{\mu,\sigma_S}$ is $\mathscr{T}_{\mu,\sigma_S} = \mathscr{K}_\mu + \mathscr{B}_{\mathbf{T}}(\sigma_S)$, where $\mathscr{B}_{\mathbf{T}}(\sigma_S) \subset S_d$ is the closed ball of traceless symmetric tensors with norm lower than σ_S.

1.3.3 Dilatancy and non-associated Drucker–Prager flow rule

Drucker and Prager (1952) deduce from the theory of GSM that, using their yield surface as a dissipation potential, "volume expansion is seen to be a necessary accompaniment to shearing deformation". As mentioned in our introduction, such a phenomenon is known as dilatancy.

Indeed, we can easily verify that the associated Drucker–Prager flow rule predicts a positive dilatancy. Defining a plasticity potential from the Drucker–Prager yield surface, $\mathscr{D}^\star(\boldsymbol{\sigma}) = \mathscr{I}_{-\mathscr{T}_{\mu,\sigma_S}}(\boldsymbol{\sigma} - \tau_c \boldsymbol{\iota}_d)$, one gets that

$$\dot{\boldsymbol{\varepsilon}} \in \frac{\partial \mathscr{D}^\star}{\partial \boldsymbol{\sigma}} \iff \dot{\boldsymbol{\varepsilon}} \in \mathscr{N}_{-\mathscr{T}_{\mu,\sigma_S}}(\boldsymbol{\sigma} - \tau_c \boldsymbol{\iota}_d).$$

Now if $\sigma_S = 0$, the normal cone to $\mathscr{T}_{\mu,\sigma_S}$ obviously coincide with that of $\mathscr{K}_\mu$, and for $\sigma_S > 0$,

$$\dot{\boldsymbol{\varepsilon}} \in \mathscr{N}_{-\mathscr{T}_{\mu,\sigma_S}}(\boldsymbol{\sigma}) \iff \begin{cases} \boldsymbol{\sigma} \in -\mathscr{T}_{\mu,\sigma_S} & \\ \dot{\boldsymbol{\varepsilon}} = \mathbf{0} & \text{if } \boldsymbol{\sigma} \in \operatorname{int} -\mathscr{T}_{\mu,\sigma_S} \\ \dot{\boldsymbol{\varepsilon}}_{\mathrm{N}} > 0 \text{ and } \dot{\boldsymbol{\varepsilon}}_{\mathbf{T}} = \mathbf{0} & \text{if } \boldsymbol{\sigma}_{\mathrm{N}} = 0 \text{ and } |\boldsymbol{\sigma}_{\mathbf{T}}| < \sigma_S \\ \dot{\boldsymbol{\varepsilon}} \in \mathscr{K}_{\frac{1}{\mu}} \cap \{\boldsymbol{\sigma} - \sigma_S \frac{\boldsymbol{\sigma}_{\mathbf{T}}}{|\boldsymbol{\sigma}_{\mathbf{T}}|}\}^\perp & \text{if } \boldsymbol{\sigma}_{\mathrm{N}} > 0 \text{ and } |\boldsymbol{\sigma}_{\mathbf{T}}| = \sigma_S - \mu \boldsymbol{\sigma}_{\mathrm{N}} \\ \dot{\boldsymbol{\varepsilon}} \in \mathscr{K}_{\frac{1}{\mu}} \cap \left\{\alpha \boldsymbol{\iota}_d + \beta \frac{\boldsymbol{\sigma}_{\mathbf{T}}}{|\boldsymbol{\sigma}_{\mathbf{T}}|},\ \alpha \in \mathbb{R}, \beta \in \mathbb{R}_+\right\} & \text{if } \boldsymbol{\sigma}_{\mathrm{N}} = 0 \text{ and } |\boldsymbol{\sigma}_{\mathbf{T}}| = \sigma_S. \end{cases} \tag{1.18}$$

In any case, we see that

$$\dot{\boldsymbol{\varepsilon}} \in \mathscr{N}_{-\mathscr{T}_{\mu,\sigma_S}}(\boldsymbol{\sigma} - \tau_c \boldsymbol{\iota}_d) \implies \dot{\boldsymbol{\varepsilon}} \in \mathscr{K}_{\frac{1}{\mu}}.$$

The associated flow rule therefore implies that the norm of the shear rate is limited by the normal part of the strain rate: $\zeta |\boldsymbol{\sigma}_{\mathbf{T}}| \leq \boldsymbol{\sigma}_{\mathrm{N}}$ with the dilatancy coefficient ζ equal to the friction coefficient μ. This translates to the dilatancy angle ψ being equal to the friction angle φ.

Dilatancy makes physical sense; indeed, a grain initially stuck between others, as in Figure 1.6(a), will first have to go upwards before being able to perform any lateral motion. However, we see from the above computation that a lower friction coefficient will mean a lower dilatancy, which is not necessarily intuitive. Actually, experiments on granular materials disproved this postulate, and the dilatancy was observed to be significantly lower than predicted by the associated flow rule (see e.g., Vermeer 1998). Moreover, the same argument that we used to reject an associated flow rule for the Coulomb friction law can be made again: for a cohesionless material, $\sigma_S = \tau_c = 0$, the rate of energy dissipated through the plastic strain rate, $\dot{\boldsymbol{\varepsilon}} : \boldsymbol{\sigma}$, would be always zero.

Overall, this motivates keeping the dilatancy angle ψ, or equivalently our dilatancy coefficient θ, as a independent parameter of the model. Noting that $\dot{\boldsymbol{\varepsilon}} \in \mathscr{K}_{\frac{1}{\zeta}} \iff \dot{\boldsymbol{\varepsilon}} + (\mu - \zeta)\,|\boldsymbol{\varepsilon}_{\mathbf{T}}|\boldsymbol{\iota}_d \in \mathscr{K}_{\frac{1}{\mu}}$, we will follow Berga and de Saxcé (1994) and use the non-associated flow rule defined by

$$\dot{\boldsymbol{\varepsilon}} + (\mu - \zeta)\,|\dot{\boldsymbol{\varepsilon}}_{\mathbf{T}}|\boldsymbol{\iota}_d \in \mathscr{N}_{-\mathscr{T}_{\mu,\sigma_S}}(\boldsymbol{\sigma} - \tau_c \boldsymbol{\iota}_d). \tag{1.19}$$

For the material to be stable, plastic displacements need to dissipate energy, which means $\dot{\boldsymbol{\varepsilon}} : \boldsymbol{\sigma} \geq 0$ for all admissible stresses $\boldsymbol{\sigma}$. This implies $0 \leq \zeta \leq \mu$, or in terms of angles, $0 \leq \psi \leq \phi$. Note that for $\zeta = \mu$, we retrieve the associated case; however, for $\zeta < \mu$ we leave the GSM framework and obtain an Implicit Standard Material. For $\zeta = 0$, and a cohesionless material, we retrieve the non-associated Coulomb friction flow rule. Finally, for $\zeta = \mu = 0$, $\sigma_S > 0$ and $\tau_c = +\infty$, we obtain the incompressible Bingham yield surface (an infinite cylinder around the hydrostatic axis) with associated flow rule; for $\tau_c = 0$, we get unilateral incompressibility, that is, plastic expansion is allowed but not compression. More generally, we will see in the following section that the flow rule (1.19) enforces a maximum dissipation principle on the deviatoric component of the stress.

In order to lighten notations, we will denote by $\mathscr{DP}(\mu, \sigma_S, \tau_c, \zeta)$ the set of $(\dot{\boldsymbol{\varepsilon}}, -\boldsymbol{\sigma}) \in S_d \times S_d$ such that the flow rule (1.19) is satisfied. We put the "minus" in front of the stress variable in order to be consistent with our definition of the Coulomb friction solution set $\mathscr{C}_\mu$. Indeed we recognize from Property 1.4 that $\mathscr{C}_\mu \sim \mathscr{DP}(\mu, 0, 0, 0)$, the only difference being that the former is defined on vector spaces, while the later is defined on tensor spaces. As such, the flow rule (1.19) naturally motivates the introduction of another version of the *de Saxcé change of variable* presented in Property 1.4, and illustrated in Figure 1.6(b).

Property 1.6. *With the change of variable $\dot{\boldsymbol{\varepsilon}} \mapsto \tilde{\dot{\boldsymbol{\varepsilon}}} := \dot{\boldsymbol{\varepsilon}} + (\mu - \zeta)\,|\dot{\boldsymbol{\varepsilon}}_T|$, the Drucker–Prager flow rule may be written as*

$$(\dot{\boldsymbol{\varepsilon}}, -\boldsymbol{\sigma}) \in \mathscr{DP}(\mu, \sigma_S, \tau_c, \zeta) \iff \tilde{\dot{\boldsymbol{\varepsilon}}} \in \mathscr{N}_{-\mathscr{T}_{\mu,\sigma_S}}(\boldsymbol{\sigma} - \tau_c \boldsymbol{\iota}_d).$$

When there are no ambiguities, we will omit the parameters which are taken to be zero, e.g.,, $\mathscr{DP}(\mu, \tau_c) = \mathscr{DP}(\mu, 0, \tau_c, 0)$. Indeed, a non-zero τ_c do not fundamentally change the nature of the solution set, so we will often discard it.

1.3.4 Bipotential and reformulations of the Drucker–Prager flow rule

Using the bipotential framework, we will show that $\mathscr{DP}(\mu, \sigma_S, \tau_c, \zeta)$ is fully characterized by the disjunctive formulation (1.20),

$$\begin{cases} \boldsymbol{\sigma}_{\mathbf{T}} = (\sigma_S + \mu(\tau_c - \boldsymbol{\sigma}_{\mathrm{N}}))\,\dfrac{\dot{\boldsymbol{\varepsilon}}_{\mathbf{T}}}{|\dot{\boldsymbol{\varepsilon}}_{\mathbf{T}}|} & \text{if } \dot{\boldsymbol{\varepsilon}}_{\mathbf{T}} \neq \mathbf{0} \\ |\boldsymbol{\sigma}_{\mathbf{T}}| \leq (\sigma_S + \mu(\tau_c - \boldsymbol{\sigma}_{\mathrm{N}})) & \text{if } \dot{\boldsymbol{\varepsilon}}_{\mathbf{T}} = \mathbf{0} \\ \quad 0 \leq \tau_c - \boldsymbol{\sigma}_{\mathrm{N}} \perp \dot{\boldsymbol{\varepsilon}}_{\mathrm{N}} - \zeta|\dot{\boldsymbol{\varepsilon}}_{\mathbf{T}}| \geq 0. \end{cases} \tag{1.20}$$

We mentioned in the previous paragraph that for $\sigma_S = \tau_c = \zeta = 0$, we retrieved a flow rule similar to that of the Coulomb contact law; that is, $\mathscr{C}_\mu \sim \mathscr{DP}(\mu)$. This motivates looking for a bipotential with a structure similar to Equation (1.14), but adapted to our new version of the *de Saxcé* change of variable from Property 1.6.

Property 1.7 (Bipotential for the Drucker–Prager flow rule). *The function* $b : S_d \times S_d \to \bar{\mathbb{R}}$,

$$b(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}) = \mathscr{I}_{-\mathscr{T}_{\mu,\sigma_S}}(\boldsymbol{\sigma} - \tau_c \boldsymbol{\iota}_d) + \mathscr{I}_{\mathscr{K}_{\frac{1}{\theta}}}(\dot{\boldsymbol{\varepsilon}}) + \tau_c \dot{\boldsymbol{\varepsilon}}_N + \sigma_S |\dot{\boldsymbol{\varepsilon}}_T| + (\mu - \zeta)(\tau_c - \boldsymbol{\sigma}_N)|\dot{\boldsymbol{\varepsilon}}_T|$$

is a bipotential modeling the non-associated Drucker–Prager flow rule (1.19). That is,

$$\begin{aligned}(\dot{\boldsymbol{\varepsilon}}, -\boldsymbol{\sigma}) \in \mathscr{DP}(\mu, \sigma_S, \tau_c, \zeta) &\iff b(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}) = \langle \dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma} \rangle \\ &\iff \dot{\boldsymbol{\varepsilon}} \in \frac{\partial b}{\partial \boldsymbol{\sigma}}(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}) \iff \boldsymbol{\sigma} \in \frac{\partial b}{\partial \dot{\boldsymbol{\varepsilon}}}(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}) \\ &\iff \textit{the disjunctive formulation (1.20) is satisfied.}\end{aligned}$$

Proof. It follows from the definition of $\mathscr{DP}$ and the Drucker–Prager flow rule (1.19) that $(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}) \in \mathscr{DP}(\mu, \sigma_S, \tau_c, \zeta) \iff \dot{\boldsymbol{\varepsilon}} \in \frac{\partial b}{\partial \boldsymbol{\sigma}}(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma})$.

Let us now show that b is a bipotential. b is the sum of convex functions w.r.t. $\boldsymbol{\sigma}$ and $\dot{\boldsymbol{\varepsilon}}$, and is therefore is convex w.r.t. each variable; indeed, we supposed that $\mu \geq \zeta$, and $\boldsymbol{\sigma}_N \leq \tau_c$ everywhere on the effective domain of b. Moreover, for any $\boldsymbol{\sigma}$ and $\dot{\boldsymbol{\varepsilon}}$ in the effective domain of b, we have $\dot{\boldsymbol{\varepsilon}}_N \geq \zeta |\dot{\boldsymbol{\varepsilon}}_T|$, and $\boldsymbol{\sigma}_T \leq \mu(\tau_c - \boldsymbol{\sigma}_N) + \sigma_S$, therefore

$$\begin{aligned}\langle \dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma} \rangle &= \dot{\boldsymbol{\varepsilon}}_N \boldsymbol{\sigma}_N + \langle \dot{\boldsymbol{\varepsilon}}_T, \boldsymbol{\sigma}_T \rangle \leq \dot{\boldsymbol{\varepsilon}}_N \tau_c + \dot{\boldsymbol{\varepsilon}}_N (\boldsymbol{\sigma}_N - \tau_c) + |\dot{\boldsymbol{\varepsilon}}_T||\boldsymbol{\sigma}_T| \\ &\leq \dot{\boldsymbol{\varepsilon}}_N \tau_c + \zeta |\dot{\boldsymbol{\varepsilon}}_T|(\boldsymbol{\sigma}_N - \tau_c) + |\dot{\boldsymbol{\varepsilon}}_T|(\mu(\tau_c - \boldsymbol{\sigma}_N) + \sigma_S) \\ &\leq \tau_c \dot{\boldsymbol{\varepsilon}}_N + \sigma_S |\dot{\boldsymbol{\varepsilon}}_T| + (\mu - \zeta)(\tau_c - \boldsymbol{\sigma}_N)|\dot{\boldsymbol{\varepsilon}}_T| \\ &\leq b(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}).\end{aligned}$$

Therefore, b is a bipotential.

From Property 1.2, this means that we only have to show that

$$\begin{cases} (\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}) \in \mathscr{DP}(\mu, \sigma_S, \tau_c, \zeta) \implies b(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}) = \langle \dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma} \rangle & (1.21) \\ (1.20) \implies b(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}) = \langle \dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma} \rangle & (1.22) \\ (1.20) \iff \boldsymbol{\sigma} \in \frac{\partial b}{\partial \dot{\boldsymbol{\varepsilon}}}(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}), & (1.23) \end{cases}$$

to obtain the complete equivalence chain of Property 1.7. We can first check in each case of the expression of the normal cone of $\mathscr{T}_{\mu,\sigma_S}$ given in Equation (1.18) that the implication (1.21) holds. Let $\tilde{\dot{\boldsymbol{\varepsilon}}}$ denote the *de Saxcé* change of variable from Property 1.6 and $\tilde{\boldsymbol{\sigma}} := \boldsymbol{\sigma} - \tau_c \boldsymbol{\iota}_d$. The flow rule (1.19) means $\tilde{\dot{\boldsymbol{\varepsilon}}} \in \mathscr{N}_{-\mathscr{T}_{\mu,\sigma_S}}(\tilde{\boldsymbol{\sigma}})$, and then either:

1. $\tilde{\boldsymbol{\sigma}} \in \operatorname{int} -\mathscr{T}_{\mu,\sigma_S}$, then $\tilde{\dot{\boldsymbol{\varepsilon}}} = \dot{\boldsymbol{\varepsilon}} = \mathbf{0}$, therefore $b(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}) = \langle \dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma} \rangle = 0$
2. $\tilde{\boldsymbol{\sigma}}_T < \sigma_S$ and $\tilde{\boldsymbol{\sigma}}_N = 0$, then $\dot{\boldsymbol{\varepsilon}}_T = \mathbf{0}$ and $b(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}) = \dot{\boldsymbol{\varepsilon}}_N \tau_c = \langle \dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma} \rangle$
3. $\tilde{\boldsymbol{\sigma}}_N > 0$ and $|\boldsymbol{\sigma}_T| = \sigma_S - \mu \tilde{\boldsymbol{\sigma}}_N$, then

$$\left\langle \tilde{\dot{\boldsymbol{\varepsilon}}}, \tilde{\boldsymbol{\sigma}} \right\rangle = \left\langle \tilde{\dot{\boldsymbol{\varepsilon}}}, \tilde{\boldsymbol{\sigma}} - \sigma_S \frac{\boldsymbol{\sigma}_T}{|\boldsymbol{\sigma}_T|} \right\rangle + \sigma_S \left\langle \tilde{\dot{\boldsymbol{\varepsilon}}}, \frac{\boldsymbol{\sigma}_T}{|\boldsymbol{\sigma}_T|} \right\rangle = \sigma_S |\dot{\boldsymbol{\varepsilon}}_T|$$

and

$$\begin{aligned}\langle \dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma} \rangle &= \left\langle \tilde{\dot{\boldsymbol{\varepsilon}}}, \tilde{\boldsymbol{\sigma}} \right\rangle + \tau_c \dot{\boldsymbol{\varepsilon}}_N - \langle (\mu - \theta)|\dot{\boldsymbol{\varepsilon}}| \boldsymbol{\iota}_d, \tilde{\boldsymbol{\sigma}} \rangle \\ &= \sigma_S |\dot{\boldsymbol{\varepsilon}}_T| + \tau_c \dot{\boldsymbol{\varepsilon}}_N - (\mu - \theta)\tilde{\boldsymbol{\sigma}}_N |\dot{\boldsymbol{\varepsilon}}_T| = b(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}).\end{aligned}$$

4. $\tilde{\boldsymbol{\sigma}}_N = 0$ and $|\boldsymbol{\sigma}_T| = \sigma_S$, then once again $\left\langle \tilde{\dot{\boldsymbol{\varepsilon}}}, \tilde{\boldsymbol{\sigma}} \right\rangle = \sigma_S |\dot{\boldsymbol{\varepsilon}}_T|$, and

$$\langle \dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma} \rangle = \left\langle \tilde{\dot{\boldsymbol{\varepsilon}}}, \tilde{\boldsymbol{\sigma}} \right\rangle + \tau_c \dot{\boldsymbol{\varepsilon}}_N - \langle (\mu - \theta)|\dot{\boldsymbol{\varepsilon}}| \boldsymbol{\iota}_d, \tilde{\boldsymbol{\sigma}} \rangle = \sigma_S |\dot{\boldsymbol{\varepsilon}}_T| + \tau_c \dot{\boldsymbol{\varepsilon}}_N = b(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}).$$

Now, let us treat the right-hand side of the equivalence chain. First, we show the equivalence (1.23),

$$(1.20) \iff \boldsymbol{\sigma} \in \frac{\partial b}{\partial \dot{\boldsymbol{\varepsilon}}}(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}).$$

Indeed,

$$\begin{aligned}\boldsymbol{\sigma} \in \frac{\partial b}{\partial \dot{\boldsymbol{\varepsilon}}}(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}) &\iff \tilde{\boldsymbol{\sigma}} \in \mathcal{N}_{\mathcal{K}_{\frac{1}{\zeta}}}(\dot{\boldsymbol{\varepsilon}}) + (\sigma_S - (\mu - \zeta))\,\tilde{\boldsymbol{\sigma}}_{\mathrm{N}} \frac{\partial |\dot{\boldsymbol{\varepsilon}}_{\mathbf{T}}|}{\partial \dot{\boldsymbol{\varepsilon}}} \\ &\iff \begin{cases} \dot{\boldsymbol{\varepsilon}} \in \mathcal{K}_{\frac{1}{\zeta}} \\ \tilde{\boldsymbol{\sigma}} \in (\sigma_S - (\mu-\zeta)\tilde{\boldsymbol{\sigma}}_{\mathrm{N}}) \dfrac{\dot{\boldsymbol{\varepsilon}}_{\mathbf{T}}}{|\dot{\boldsymbol{\varepsilon}}_{\mathbf{T}}|} - \mathcal{K}_\theta \cap \{\dot{\boldsymbol{\varepsilon}}\}^\perp & \text{if } \dot{\boldsymbol{\varepsilon}}_{\mathbf{T}} \neq 0 \\ \tilde{\boldsymbol{\sigma}} \in \mathcal{B}_{\mathbf{T}}((\sigma_S - (\mu-\zeta)\tilde{\boldsymbol{\sigma}}_{\mathrm{N}})) - \mathcal{K}_\theta \cap \{\dot{\boldsymbol{\varepsilon}}\}^\perp & \text{if } \dot{\boldsymbol{\varepsilon}}_{\mathbf{T}} = 0. \end{cases} \end{aligned} \tag{1.24}$$

Now, for any $\boldsymbol{\tau} \in S_d$, $\boldsymbol{\tau} - \zeta \boldsymbol{\tau}_{\mathrm{N}} \frac{\boldsymbol{\tau}_{\mathbf{T}}}{|\boldsymbol{\tau}_{\mathbf{T}}|} \in -\mathcal{K}_\zeta \iff \boldsymbol{\tau} \in -\mathcal{K}_0$, so for $\dot{\boldsymbol{\varepsilon}} \in \mathcal{K}_{\frac{1}{\zeta}}$,

$$\boldsymbol{\tau} - \zeta \boldsymbol{\tau}_{\mathrm{N}} \frac{\boldsymbol{\tau}_{\mathbf{T}}}{|\boldsymbol{\tau}_{\mathbf{T}}|} \in -\mathcal{K}_\zeta \cap \{\dot{\boldsymbol{\varepsilon}}\}^\perp \iff \boldsymbol{\tau} \in -\mathcal{K}_0 \text{ and } \boldsymbol{\tau}_{\mathrm{N}}(\dot{\boldsymbol{\varepsilon}}_{\mathrm{N}} - \zeta |\dot{\boldsymbol{\varepsilon}}_{\mathbf{T}}|).$$

Moreover, $\boldsymbol{\tau} \in -\mathcal{K}_\zeta \cap \{\dot{\boldsymbol{\varepsilon}}\}^\perp$ implies that $\boldsymbol{\tau}_{\mathbf{T}}$ is colinear with $\dot{\boldsymbol{\varepsilon}}_{\mathbf{T}}$ (with $\dot{\boldsymbol{\varepsilon}}_{\mathbf{T}}$ eventually zero). Using these results in Equation (1.24), we get

$$\boldsymbol{\sigma} \in \frac{\partial b}{\partial \dot{\boldsymbol{\varepsilon}}}(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}) \iff \begin{cases} \dot{\boldsymbol{\varepsilon}} \in \mathcal{K}_{\frac{1}{\zeta}} \\ \tilde{\boldsymbol{\sigma}}_{\mathrm{N}} \leq 0 \\ 0 = \tilde{\boldsymbol{\sigma}}_{\mathrm{N}}(\dot{\boldsymbol{\varepsilon}}_{\mathrm{N}} - \zeta |\dot{\boldsymbol{\varepsilon}}_{\mathbf{T}}|) \\ \boldsymbol{\sigma}_{\mathbf{T}} = (\sigma_S - \mu \tilde{\boldsymbol{\sigma}}_{\mathrm{N}}) \dfrac{\dot{\boldsymbol{\varepsilon}}_{\mathbf{T}}}{|\dot{\boldsymbol{\varepsilon}}_{\mathbf{T}}|} & \text{if } \dot{\boldsymbol{\varepsilon}}_{\mathbf{T}} \neq 0 \\ \boldsymbol{\sigma}_{\mathbf{T}} \in \mathcal{B}_{\mathbf{T}}((\sigma_S - \mu \tilde{\boldsymbol{\sigma}}_{\mathrm{N}})) & \text{if } \dot{\boldsymbol{\varepsilon}}_{\mathbf{T}} = 0 \end{cases} \iff (1.20).$$

It now only remains to verify (1.21), i.e., that the disjunctive formulation (1.20) implies that $b(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}) = \langle \dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma} \rangle$. Indeed, if $\dot{\boldsymbol{\varepsilon}}$ and $\boldsymbol{\sigma}$ satisfy (1.20), we have

$$\begin{aligned} \langle \dot{\boldsymbol{\varepsilon}}_{\mathbf{T}}, \boldsymbol{\sigma}_{\mathbf{T}} \rangle &= (\sigma_S + \mu(\tau_c - \boldsymbol{\sigma}_{\mathrm{N}}))\,|\dot{\boldsymbol{\varepsilon}}_{\mathbf{T}}| \\ \dot{\boldsymbol{\varepsilon}}_{\mathrm{N}} \boldsymbol{\sigma}_{\mathrm{N}} &= \tilde{\boldsymbol{\sigma}}_{\mathrm{N}}(\dot{\boldsymbol{\varepsilon}}_{\mathrm{N}} - \zeta |\dot{\boldsymbol{\varepsilon}}_{\mathbf{T}}|) + \tilde{\boldsymbol{\sigma}}_{\mathrm{N}} \zeta |\dot{\boldsymbol{\varepsilon}}_{\mathbf{T}}| + \tau_c \dot{\boldsymbol{\varepsilon}}_{\mathrm{N}} = (\boldsymbol{\sigma}_{\mathrm{N}} - \tau_c)\zeta |\dot{\boldsymbol{\varepsilon}}_{\mathbf{T}}| + \tau_c \dot{\boldsymbol{\varepsilon}}_{\mathrm{N}}, \end{aligned}$$

so $b(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}) = \langle \dot{\boldsymbol{\varepsilon}}_{\mathbf{T}}, \boldsymbol{\sigma}_{\mathbf{T}} \rangle + \dot{\boldsymbol{\varepsilon}}_{\mathrm{N}} \boldsymbol{\sigma}_{\mathrm{N}}$.

□

Convex conjugate of the truncated SOC characteristic function Property 1.7 implicitly gives an expression for the subdifferential of ${\mathscr{I}_{\mathscr{T}_{\mu,\sigma_S}}}^\star$. Indeed, considering the Drucker–Prager flow rule for $\mu = \zeta$ and $\tau_c = 0$, we get

$$\begin{aligned} \boldsymbol{\sigma} \in \partial\left({\mathscr{I}_{-\mathscr{T}_{\mu,\sigma_S}}}^\star\right)(\dot{\boldsymbol{\varepsilon}}) &\iff \dot{\boldsymbol{\varepsilon}} \in \mathcal{N}_{-\mathscr{T}_{\mu,\sigma_S}}(\boldsymbol{\sigma}) \\ &\iff \boldsymbol{\sigma} \in \frac{\partial b}{\partial \dot{\boldsymbol{\varepsilon}}}(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}) \iff \boldsymbol{\sigma} \in \partial\left(\mathscr{I}_{\mathcal{K}_{\frac{1}{\mu}}} + \sigma_S |\boldsymbol{\sigma}_{\mathbf{T}}|\right). \end{aligned}$$

As moreover ${\mathscr{I}_{-\mathscr{T}_{\mu,\sigma_S}}}^\star(\mathbf{0}) = \left(\mathscr{I}_{\mathcal{K}_{\frac{1}{\mu}}} + \sigma_S |\boldsymbol{\sigma}_{\mathbf{T}}|\right) = \mathbf{0}$, we deduce the equality (Moreau 1966–1967, Proposition 10.j)

$${\mathscr{I}_{-\mathscr{T}_{\mu,\sigma_S}}}^\star = \mathscr{I}_{\mathcal{K}_{\frac{1}{\mu}}} + \sigma_S |\boldsymbol{\sigma}_{\mathbf{T}}|. \tag{1.25}$$

Maximum dissipation principle The first two equations of the disjunctive formulation (1.20) may be rewritten in an equivalent manner as $\boldsymbol{\sigma}_{\mathbf{T}} \in \mathcal{N}_{\mathcal{B}_{\mathbf{T}}(\sigma_S + (\tau_c - \boldsymbol{\sigma}_{\mathrm{N}})\mu)}(\dot{\boldsymbol{\varepsilon}}_{\mathbf{T}})$, which we recognize from Theorem A.6 as the optimality conditions of the optimization problem

$$\max_{\boldsymbol{\tau} \in \mathcal{B}_{\mathbf{T}}(\sigma_S + (\tau_c - \boldsymbol{\sigma}_{\mathrm{N}})\mu)} \langle \boldsymbol{\tau}, \dot{\boldsymbol{\varepsilon}}_{\mathbf{T}} \rangle.$$

In other words, $\boldsymbol{\sigma}_{\mathbf{T}}$ maximizes the dissipated energy over the set of admissible deviatoric stresses.

Complementarity functions We can use the disjunctive formulation (1.20) to define an extension of the Alart–Curnier complementarity function (1.6) whose roots will coincide with the solutions of the Drucker–Prager flow rule,

$$(\dot{\boldsymbol{\varepsilon}}, -\boldsymbol{\sigma}) \in \mathscr{D}\mathscr{P}(\mu, \sigma_S, \tau_c, \zeta) \iff f_{\mathrm{AC}}(\dot{\boldsymbol{\varepsilon}}, -\boldsymbol{\sigma}) = \mathbf{0} \tag{1.26}$$

with, for any $\xi \in \mathbb{R}_+^*$,

$$\begin{array}{rccl} f_{\mathrm{AC}}: & S_d \times S_d & \to & S_d \\ & (\dot{\boldsymbol{\varepsilon}}, -\boldsymbol{\sigma}) & \mapsto & \begin{pmatrix} \Pi_{\mathbb{R}_+}(\tau_c - \boldsymbol{\sigma}_{\mathrm{N}} - \xi(\dot{\boldsymbol{\varepsilon}}_{\mathrm{N}} - \theta|\dot{\boldsymbol{\varepsilon}}_{\mathbf{T}}|)) \\ \Pi_{\mathscr{B}_{\mathbf{T}}(\sigma_S + \mu(\tau_c - \boldsymbol{\sigma}_{\mathrm{N}}))}(-\boldsymbol{\sigma}_{\mathbf{T}} - \xi\dot{\boldsymbol{\varepsilon}}_{\mathbf{T}}) \end{pmatrix} + \boldsymbol{\sigma} - \tau_c \boldsymbol{\iota}_d. \end{array} \tag{1.27}$$

In both the disjunctive formulation (1.20) and the Alart–Curnier formulation (1.27), a non-zero dilatancy significantly complicates the model, yielding a nonlinear complementarity problem on the normal components of $\dot{\boldsymbol{\varepsilon}}$ and $\boldsymbol{\sigma}$. However, this coefficient has not as much influence on the structure of the bipotential. This motivates tackling problems with non-zero dilatancy with methods derived from the bipotential framework, for instance using the *de Saxcé* change of variable. As such, we can define the De Saxcé complementarity function f_{DS} satisfying

$$(\dot{\boldsymbol{\varepsilon}}, -\boldsymbol{\sigma}) \in \mathscr{D}\mathscr{P}(\mu, \sigma_S, \tau_c, \zeta) \iff f_{\mathrm{DS}}(\dot{\boldsymbol{\varepsilon}}, -\boldsymbol{\sigma}) = \mathbf{0} \tag{1.28}$$

with, for any $\xi \in \mathbb{R}_+^*$,

$$\begin{array}{rccl} f_{\mathrm{DS}}: & S_d \times S_d & \to & S_d \\ & (\dot{\boldsymbol{\varepsilon}}, -\boldsymbol{\sigma}) & \mapsto & \Pi_{\mathscr{T}_{\mu,\sigma_S}}\left(\tau_c \boldsymbol{\iota}_d - \boldsymbol{\sigma} - \xi\tilde{\dot{\boldsymbol{\varepsilon}}}\right) + \boldsymbol{\sigma} - \tau_c \boldsymbol{\iota}_d \end{array} \tag{1.29}$$

and $\tilde{\dot{\boldsymbol{\varepsilon}}}$ defined as the *de Saxcé* change of variable of Property 1.6, $\tilde{\dot{\boldsymbol{\varepsilon}}} := \dot{\boldsymbol{\varepsilon}} + (\mu - \zeta)|\dot{\boldsymbol{\varepsilon}}_{\mathbf{T}}|\boldsymbol{\iota}_d$.

1.3.5 Viscoplasticity

Starting again from the GSM equations, we can model viscoplasticity by waiving the assumption that $\mathscr{D}$ depends only on the plastic displacement $\dot{\boldsymbol{\varepsilon}}_p$, and writing instead $\mathscr{D} = \mathscr{V}(\dot{\boldsymbol{\varepsilon}}) + \mathscr{D}_p(\dot{\boldsymbol{\varepsilon}}_p)$. Equations (1.7–1.8) become

$$\begin{cases} \boldsymbol{\sigma} \in \dfrac{\partial \mathscr{V}}{\partial \dot{\boldsymbol{\varepsilon}}}(\dot{\boldsymbol{\varepsilon}}) + \dfrac{\partial \mathscr{D}_p}{\partial \dot{\boldsymbol{\varepsilon}}_p}(\dot{\boldsymbol{\varepsilon}}_p) \\ \boldsymbol{\sigma} \in \dfrac{\partial \mathscr{V}}{\partial \dot{\boldsymbol{\varepsilon}}}(\dot{\boldsymbol{\varepsilon}}) + \rho \dfrac{\partial \mathscr{E}}{\partial \boldsymbol{\varepsilon}_e}(\boldsymbol{\varepsilon}_e). \end{cases}$$

With the viscous dissipation potential $\mathscr{V}(\dot{\boldsymbol{\varepsilon}}) = \eta_N \dot{\boldsymbol{\varepsilon}}_{\mathrm{N}}^2 + \eta_T \dot{\boldsymbol{\varepsilon}}_{\mathbf{T}}^2$, we get

$$\begin{cases} \boldsymbol{\sigma} = \boldsymbol{\sigma}_e + (\eta_T \dot{\boldsymbol{\varepsilon}} + (\eta_N - \eta_T)\dot{\boldsymbol{\varepsilon}}_{\mathrm{N}} \boldsymbol{\iota}_d) \\ \boldsymbol{\sigma}_e \in \dfrac{\partial \mathscr{D}_p}{\partial \dot{\boldsymbol{\varepsilon}}_p}(\dot{\boldsymbol{\varepsilon}}_p) \\ \boldsymbol{\sigma}_e \in \rho \dfrac{\partial \mathscr{E}}{\partial \boldsymbol{\varepsilon}_e}(\boldsymbol{\varepsilon}_e). \end{cases}$$

We can once again generalize this equations to the framework of Implicit Standard Materials by allowing $\mathscr{D}_p$ to depend on $\boldsymbol{\sigma}_e$ as well as on $\dot{\boldsymbol{\varepsilon}}_p$. Our full visco-elasto-plastic model then reads

$$\begin{cases} \boldsymbol{\sigma} = \boldsymbol{\sigma}_e + \left(\eta_T\left(\dot{\boldsymbol{\varepsilon}}_e + \dot{\boldsymbol{\varepsilon}}_p\right) + (\eta_N - \eta_T)\left(\dot{\boldsymbol{\varepsilon}}_e + \dot{\boldsymbol{\varepsilon}}_p\right)_{\mathrm{N}} \boldsymbol{\iota}_d\right) \\ \boldsymbol{\sigma}_e \in \dfrac{\partial b}{\partial \dot{\boldsymbol{\varepsilon}}_p}(\dot{\boldsymbol{\varepsilon}}_p, \boldsymbol{\sigma}_e) \\ \boldsymbol{\sigma}_e \in \rho \dfrac{\partial \mathscr{E}}{\partial \boldsymbol{\varepsilon}_e}(\boldsymbol{\varepsilon}_e). \end{cases} \tag{1.30}$$

In this thesis, we will restrict ourselves to the case where $\mathscr{E} = \mathscr{I}_{\mathbf{0}}$, i.e., the viscoplastic case where $\boldsymbol{\varepsilon}_e = 0$, and will choose b to be the Drucker–Prager bipotential defined in Property 1.7.

Summary

In this first technical chapter, we have derived several equivalent formulations of the Signorini–Coulomb contact law (1–3) and of the non-associated Drucker–Prager flow rule with dilatancy (1.19), which shall prove useful for their numerical resolution. We have seen that these laws do not fit into the framework of Generalized Standard Materials, but can be described using the more comprehensive theory of Implicit Standard Materials.

Other cones In this chapter, we studied the nature of laws that were intrinsically linked to the Second-Order Cone (SOC). It is natural to wonder if and how the equivalences that we obtained would translate for other kind of convex cones; for instance, the Mohr-Coulomb yield surface.

As a first step, we can define the p-order cone of aperture μ, $K_\mu^{(p)}$, as

$$K_\mu^{(p)} := \left\{ \boldsymbol{x} \in \mathbb{R}^d, \mu \boldsymbol{x}_{\mathrm{N}} \geq \|\boldsymbol{x}_{\mathrm{T}}\|_p \right\}.$$

For $p = 1$ or $p = +\infty$, the p-order cone becomes a pyramid, which yields a popular way of approximating the Coulomb law in numerical solvers, as in e.g., (Klarbring 1987). Its dual cone is given by $(K_\mu^{(p)})^\star = K_{\frac{1}{\mu}}^{(q)}$, with q such that $\frac{1}{p} + \frac{1}{q} = 1$. The *de Saxcé* change of variable must then be adapted to $\tilde{\boldsymbol{u}}^{(p)} := \boldsymbol{u} + \mu \|\boldsymbol{u}_{\mathrm{T}}\|_q \boldsymbol{n}$ so that the p-order Coulomb friction law reads $\tilde{\boldsymbol{u}}^{(p)} \in -\mathcal{N}_{K_\mu^{(p)}}(\boldsymbol{r})$, or equivalently $K_{\frac{1}{\mu}}^{(q)} \ni \tilde{\boldsymbol{u}}^{(p)} \perp \boldsymbol{r} \in K_\mu^{(p)}$.

Remaining equations The set of equations that we obtained is not sufficient to fully determine a dynamical system. For the discrete contact mechanics, we have two variables, $\boldsymbol{u}$ and $\boldsymbol{r}$, but only one equation linking them. Similarly, our visco-elasto-plastic model (1.30) boasts four variables ($\boldsymbol{\sigma}_t$,$\boldsymbol{\sigma}_e$, $\dot{\boldsymbol{\varepsilon}}_p$ and $\dot{\boldsymbol{\varepsilon}}_e$), but only three equations. We therefore need one more relationship; it will be provided the so-called conservation equations, which describe how a local stress affects the global motion of the dynamical system.

In the next chapter, we will see how to integrate the Coulomb contact law within the framework of Discrete-Element Modeling. Starting from chapter 6, we will go back to our viscoplastic model (1.30), and apply it to the context of Finite-Element Modeling of granular flows.

2 Modeling contacts within the Discrete Element Method

In this chapter, we will first briefly present a few models of mechanical systems, and write their unconstrained dynamics. Then, we will see how we can account for self or external unilateral contacts with friction.

2.1 A few mechanical models for rigid and deformable bodies in finite dimension

Before going on with more general models, let us start with the simple case of a rigid-body,

2.1.1 Rigid-body dynamics

The kinematics of a rigid-body Ω can be entirely described by a *frame* $\mathbf{Q}$; that is, the combination of a world position $\boldsymbol{p}$ and a rotation matrix R.

Rotations Formally, rotations matrices in dimension d are the elements of $\mathcal{SO}(d)$, the group of orthonormal matrices with positive determinant. However, such description is not very convenient in numerical programs; $d \times d$ coefficients have to be stored, compositing rotations requires computing expensive matrix–matrix products, and complex constraints, prone to drift, have to be enforced to ensure that the matrices stay in $\mathcal{SO}(d)$. It is thus preferable to use lower-dimensional parameterizations of the set of rotations matrices.

In 2D, the rotation matrix R can be represented simply with a scalar angle θ;

$$R = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} = \exp\begin{pmatrix} 0 & \theta \\ -\theta & 0 \end{pmatrix}.$$

Conversely, Euler's rotation theorem states that rotations in 3D can be described by the combination of an axis $\boldsymbol{a}$ (a unit vector in $\mathbb{R}^3$) and an angle θ. This decomposition is non-unique, since the angle can be chosen modulo π, and the axis can be chosen arbitrarily when $\theta = 0$ or $\theta = \pi$. More precisely, using Rodrigues's formula, every rotation matrix in $3D$ can be expressed

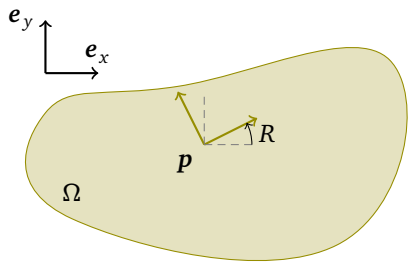


Figure 2.1: *A rigid body Ω with center of mass $\boldsymbol{p}$ and rotation R w.r.t. the world's reference frame*

as $R = \exp[\boldsymbol{e}_\wedge]$, where $[\boldsymbol{e}_\wedge]$ is the skew-symmetric matrix corresponding to the application of the cross product $\boldsymbol{e} \wedge \cdot$,

$$[\boldsymbol{e}_\wedge] := \begin{pmatrix} 0 & -\boldsymbol{e}_z & \boldsymbol{e}_y \\ \boldsymbol{e}_z & 0 & -\boldsymbol{e}_x \\ -\boldsymbol{e}_y & \boldsymbol{e}_x & 0 \end{pmatrix}.$$

The angle of the rotation R is then $\theta = \|\boldsymbol{e}\|$, and the axis is $\frac{\boldsymbol{e}}{\|\boldsymbol{e}\|}$ (or any unit vector when $\boldsymbol{e} = \boldsymbol{0}$). Alternatively, the set of 3D rotations can be parameterized by the three angles of the successive rotations around each axis of an orthogonal basis — the so-called Euler angles. However, while very compact, those representations are not much more satisfying for a numerical implementation; they require numerous trigonometric operations which are computationally expensive and may accumulate errors, and compositing rotations is still non-trivial. Euler angles are also subject to gimbal-locking, that is, in certain configurations they may loose a degree of freedom.

A better way of representing the matrix R in a compact yet well-defined manner is through the mean of a unit quaternion $\boldsymbol{q} \in \mathscr{H}_1$, $\mathscr{H}_1 = \{\boldsymbol{q} := [w,x,y,z] := w + x\boldsymbol{i} + y\boldsymbol{j} + x\boldsymbol{z}, (w,x,y,z) \in \mathbb{R}^4, \|\boldsymbol{q}\| = 1\}$ with $\boldsymbol{i}$, $\boldsymbol{j}$ and $\boldsymbol{k}$ square roots of -1. Numerous texts have been written about the relationship between unit quaternions, matrix exponentials and rotations, we refer the reader to e.g., (Cadoux 2009, Appendix C) for a concise introduction focused on their application to rigid-body dynamics. In practice, a unit quaternion $\boldsymbol{q}$ can also be seen as the combination of an angle θ and a unit direction vector $\boldsymbol{a}$. Indeed, $\boldsymbol{q} = \left[\cos(\frac{\theta}{2}); \sin(\frac{\theta}{2})\boldsymbol{a}\right]$ is unitary and reciprocally, every unit quaternion can be decomposed in such a way: for instance, $\theta = 2\,\mathrm{atan2}(\|[x,y,z]^\intercal\|, w)$, and $\boldsymbol{a} = \frac{[x,y,z]^\intercal}{\|[x,y,z]^\intercal\|}$ if $\sin\theta \neq 0$, or any other unit vector otherwise. The decomposition is therefore not unique, but the rotation defined by the axis–angle couple is unique.

Now, the major advantage of using quaternions instead of rotation matrices or their angle–axis representation is that operations on quaternions are well-suited for numerical computations. Indeed, $\mathscr{H}_1$ is a group for the canonical quaternion product $\cdot \times \cdot$, and $\boldsymbol{q}_1 \times \boldsymbol{q}_2$ coincide with the composition of the rotations represented by $\boldsymbol{q}_1$ and $\boldsymbol{q}_2$. At the same time, it is much cheaper and precise to compute than a matrix–matrix product, and less prone to numerical drift (even though the resulting quaternion must be kept unitary). Moreover, the application $\mathscr{H}_1 \times \mathbb{R}^3 \to \mathbb{R}^3$, $(\boldsymbol{q}, \boldsymbol{p}) \mapsto \boldsymbol{r}$ such that

$$[0; \boldsymbol{r}] = \boldsymbol{q}^\star \times [0; \boldsymbol{p}] \times \boldsymbol{q} \quad \text{with } ([w,x,y,z]^\intercal)^\star := [w,-x,-y,-z]^\intercal$$

coincide with the application of the rotation $R(\boldsymbol{q})$ represented by the quaternion $\boldsymbol{q}$ to the vector $\boldsymbol{p}$, i.e., $\boldsymbol{r} = R(\boldsymbol{q})\boldsymbol{p}$. Compared to Rodrigues formula, this does not involve any trigonometric operation, and is therefore much cheaper to compute numerically. Finally, unit quaternions can also be expressed as the exponential (w.r.t. the quaternion product) of a purely imaginary quaternion,

$$\left[\cos(\frac{\theta}{2}); \sin(\frac{\theta}{2})\boldsymbol{a}\right] = \exp[0; \frac{\theta}{2}\boldsymbol{a}], \tag{2.1}$$

which naturally links them to the solutions of first-order linear differential equations, and yields an insight on their integration through time.

Equations of motions Without loss of generality, we will assume that our kinematic frame $\boldsymbol{Q} = (\boldsymbol{p}; \boldsymbol{q})$ is positioned at the center of mass of the rigid body. Let also $\boldsymbol{v}$ and $\boldsymbol{\omega}$ be the linear and angular velocity of Ω at its center of mass; the time derivative of the frame $\boldsymbol{Q} := (\boldsymbol{p}; \boldsymbol{q})$ can be expressed as $\dot{\boldsymbol{Q}} := (\boldsymbol{v}; \boldsymbol{\omega})$. Indeed, $\frac{\mathrm{d}\boldsymbol{p}}{\mathrm{d}t} = \boldsymbol{v}$, and since the velocity at any point $\boldsymbol{x}$ of Ω may be computed as $\mathbf{v}(\boldsymbol{x}) = \boldsymbol{v} + \boldsymbol{\omega} \wedge (\boldsymbol{x} - \boldsymbol{p})$, it holds that $\frac{\mathrm{d}R(\boldsymbol{q})}{\mathrm{d}t}\boldsymbol{x} = \boldsymbol{\omega} \wedge (R(\boldsymbol{q})\boldsymbol{x})$, which leads to $\frac{\mathrm{d}R(\boldsymbol{q})}{\mathrm{d}t} = [\boldsymbol{\omega}]_\wedge R(\boldsymbol{q})$. This identity transpose to the quaternion framework as a 4-dimensional first-order linear differential equation, $\frac{\mathrm{d}\boldsymbol{q}}{\mathrm{d}t} = \frac{1}{2}[0; \boldsymbol{\omega}] \times \boldsymbol{q}$. Note that if the direction of $\boldsymbol{\omega}$ is constant between t_0 and t, i.e., $\boldsymbol{\omega}(t) = \dot{\theta}(t)\boldsymbol{a}$, using (2.1) the solution in time is simply given by $\boldsymbol{q}(t) = [\cos\frac{\theta(t)}{2}; \sin\frac{\theta(t)}{2}\boldsymbol{a}] \times \boldsymbol{q}(t_0)$, with $\theta(t)$ such that $\theta(t_0) = 0$ and $\frac{\mathrm{d}\theta}{\mathrm{d}t} = \dot{\theta}$.

Supposing a Galilean frame of reference, Newton's second law over a elementary volume $\mathrm{d}\boldsymbol{x}$ around $\boldsymbol{x}$ reads

$$\rho(\boldsymbol{x})\frac{\mathrm{d}\mathbf{v}(\boldsymbol{x})}{\mathrm{d}t}\mathrm{d}\boldsymbol{x} = \mathbf{g}(\boldsymbol{x},\mathbf{v}(\boldsymbol{x}),t)\mathrm{d}\boldsymbol{x}, \tag{2.2}$$

where $\mathbf{g}$ if the volumetric force applied at each point of Ω, and $\rho(\boldsymbol{x})$ is the density of the rigid body. As $\frac{\mathrm{d}\mathbf{v}(\boldsymbol{x})}{\mathrm{d}t} = \boldsymbol{v} + \frac{\mathrm{d}\boldsymbol{\omega}}{\mathrm{d}t}\wedge(\boldsymbol{x}-\boldsymbol{p}) + \boldsymbol{\omega}\wedge(\boldsymbol{\omega}\wedge(\boldsymbol{x}-\boldsymbol{p}))$, integrating over Ω yields the conservation of linear momentum equation,

$$m\frac{\mathrm{d}\boldsymbol{p}}{\mathrm{d}t} = \boldsymbol{g}(\boldsymbol{Q},\dot{\boldsymbol{Q}},t) := \int_\Omega \mathbf{g}(t,\boldsymbol{x},\mathbf{v}(\boldsymbol{x}))\mathrm{d}\boldsymbol{x}, \tag{2.3}$$

and applying the vector product $(\boldsymbol{x}-\boldsymbol{p})\wedge\cdot$ to each side of Equation (2.2) before integrating yields the conservation of angular momentum equation,

$$I(\boldsymbol{Q})\frac{\mathrm{d}\boldsymbol{\omega}}{\mathrm{d}t} + \boldsymbol{\omega}\wedge(I(\boldsymbol{Q})\boldsymbol{\omega}) = \boldsymbol{l}(\boldsymbol{Q},\dot{\boldsymbol{Q}},t) := \int_\Omega (\boldsymbol{x}-\boldsymbol{p})\wedge\mathbf{g}(t,\boldsymbol{x},\mathbf{v}(\boldsymbol{x}))\mathrm{d}\boldsymbol{x}, \tag{2.4}$$

where m is the mass and I the inertia matrix of Ω computed as

$$\begin{aligned} m &= \int_\Omega \rho(\boldsymbol{x})\mathrm{d}\boldsymbol{x} \\ I(\boldsymbol{q}) &= \mathrm{Tr}(T(\boldsymbol{q}))\mathbb{I}_3 - T(\boldsymbol{q}) \\ T(\boldsymbol{q}) &= \int_\Omega \rho(\boldsymbol{x})(\boldsymbol{x}-\boldsymbol{p})\otimes(\boldsymbol{x}-\boldsymbol{p})\mathrm{d}\boldsymbol{x}. \end{aligned}$$

We see that the inertia matrix $I(\boldsymbol{q})$, which is computed in the world's coordinate system, depends of the orientation of the frame $\boldsymbol{Q}$. However, it can easily be deduced from the computation of the inertia matrix $I_{|\boldsymbol{Q}}$ in the frame attached to the rigid body, $I(\boldsymbol{q}) = R(\boldsymbol{q})\, I_{|\boldsymbol{Q}}\, R(\boldsymbol{q})^\intercal$.

Finally, we can group the conservation equations (2.3–2.4) into a single one,

$$M(\boldsymbol{Q})\frac{\mathrm{d}\dot{\boldsymbol{Q}}}{\mathrm{d}t} = \boldsymbol{f}(\boldsymbol{Q},\dot{\boldsymbol{Q}},t), \tag{2.5}$$

$$\text{with}\quad M(\boldsymbol{Q}) := \begin{pmatrix} m\mathbb{I}_3 & 0 \\ 0 & I(\boldsymbol{q}) \end{pmatrix} \qquad \text{and} \qquad \boldsymbol{f}(\boldsymbol{Q},\dot{\boldsymbol{Q}},t) := \begin{pmatrix} \boldsymbol{g}(\boldsymbol{Q},\dot{\boldsymbol{Q}},t) \\ \boldsymbol{l}(\boldsymbol{Q},\dot{\boldsymbol{Q}},t) - \boldsymbol{\omega}\wedge(I(\boldsymbol{Q})\boldsymbol{\omega}) \end{pmatrix}.$$

Multiple bodies Now that we know how to model the dynamics of a single rigid-body, we can extend our equations to handle more complex objects, such as articulate bodies. Let (Ω_i) be a set of $N\in\mathbb{N}$ rigid-bodies with frames $(\boldsymbol{Q}_i)$. Concatenating the degrees of freedom, velocities and forces as $\boldsymbol{Q} := [\boldsymbol{Q}_1;\ldots;\boldsymbol{Q}_N]$, $\boldsymbol{f} := [\boldsymbol{f}_1;\ldots;\boldsymbol{f}_N]$ and assembling the N mass matrices M_I into a single block-diagonal matrix $M := \mathrm{diag}(M_1,\ldots,M_N)$, the dynamics of the whole system may be written as

$$\begin{cases} M(\boldsymbol{Q})\dfrac{\mathrm{d}\dot{\boldsymbol{Q}}}{\mathrm{d}t} = \boldsymbol{f}(\boldsymbol{Q},\dot{\boldsymbol{Q}},t) + \boldsymbol{f}^{\mathrm{cst}} \\ C(\boldsymbol{Q}) = \mathbf{0}. \end{cases}$$

where $C : (\mathbb{R}^3\times\mathbb{R}^4)^m \to \mathbb{R}^n$ is a function defining constraints on the system, and $\boldsymbol{f}_i^{\mathrm{cst}}$ is the sum of the forces applied on the body i through the constraints. Now, D'Alembert's principle[1] states that the work of the constraint forces $\boldsymbol{f}_i^{\mathrm{cst}}$ vanishes during a virtual displacement $\delta\boldsymbol{Q}$ (that is, an infinitesimal displacement such that the constraints are still satisfied, i.e., such that

[1] *Traité de dynamique*, Jean le Rond d'Alembert, 1743

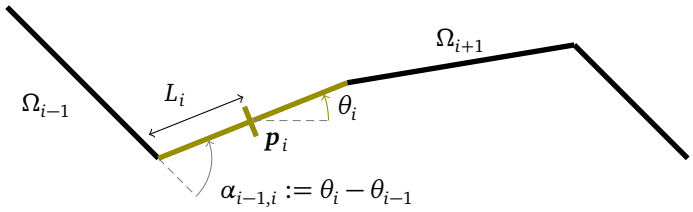


Figure 2.2: *An articulated rigid-body chain. Degrees-of-freedom are $(\boldsymbol{p}_i, \theta_i)$, and joints are modeled by holonomic constraints.*

$\delta\boldsymbol{Q} \in \mathrm{Ker}\,\frac{\partial C}{\partial \boldsymbol{Q}}(\boldsymbol{Q})$). This means that $\boldsymbol{f}^{\mathrm{cst}} \in \left(\mathrm{Ker}\,\frac{\partial C}{\partial \boldsymbol{Q}}(\boldsymbol{Q})\right)^{\perp} = \mathrm{Im}\left(\frac{\partial C}{\partial \boldsymbol{Q}}(\boldsymbol{Q})\right)^{\intercal}$, so the dynamics can be written again as

$$\begin{cases} M(\boldsymbol{Q})\dfrac{\mathrm{d}\dot{\boldsymbol{Q}}}{\mathrm{d}t} = \boldsymbol{f}(\boldsymbol{Q}, \dot{\boldsymbol{Q}}, t) + \left(\dfrac{\partial C}{\partial \boldsymbol{Q}}(\boldsymbol{Q})\right)^{\intercal} \boldsymbol{\lambda} \\ C(\boldsymbol{Q}) = \boldsymbol{0} \end{cases} \tag{2.6}$$

with $\boldsymbol{\lambda} \in \mathbb{R}^n$.

For instance, one may enforce that two bodies share the same center of mass by taking $C = (\boldsymbol{p}_1 - \boldsymbol{p}_2)$, which will lead to $\boldsymbol{f}^{cst} = (\boldsymbol{\lambda}; -\boldsymbol{\lambda})$ with $\boldsymbol{\lambda}$ in $\mathbb{R}^3$.

Chain of rigid segments As an illustrative example, we can attempt to model a 2D flexible chain of N inextensible segments of length $2L_i$ and width $r_i \ll L_i$ with frames given by $\boldsymbol{Q}_i = (\boldsymbol{p}_i, \theta_i)$, as in Figure 2.2. The end points of Ω_i are given by $\boldsymbol{p}_i \pm L\boldsymbol{e}(\theta_i)$, with $\boldsymbol{e}(\theta) := (\cos\theta; \sin\theta)$. The velocity of the point of abscissa s on the i^{th} segment is $\mathbf{v}_i(s) = \boldsymbol{v}_i + s\omega\boldsymbol{e}(\theta + \frac{\pi}{2})$. By homogeneity with the 3D equations, we will reuse the notation $\boldsymbol{x} \wedge \boldsymbol{y}$ for the determinant of two vectors in $\mathbb{R}^2$, $\boldsymbol{x} \wedge \boldsymbol{y} := \det(\boldsymbol{x}, \boldsymbol{y})$.

Assuming that the first link is clamped at the world's frame origin, and that the chain's end is free to move, our nonlinear constraint function C could be defined as

$$C(\boldsymbol{Q}) = \begin{pmatrix} \boldsymbol{p}_1 \\ \boldsymbol{p}_1 + L_1\cos\theta_1 - \boldsymbol{p}_2 + L_2\cos\theta_2 \\ \cdots \\ \boldsymbol{p}_{N-1} + L_{N-1}\cos\theta_{N-1} - \boldsymbol{p}_N + L_N\cos\theta_N \end{pmatrix}.$$

Next, assuming a constant density ρ_i over the volume of the i^{th} segment, the mass of each cylinder is $m_i = 2\rho_i L_i r_i$. In $2D$, the inertia matrix I_i boils down to a single scalar coefficient, in our case

$$I_i = \rho_i r_i \int_{-L_i}^{L_i} (s\boldsymbol{e}(\theta_i)) \wedge (s\boldsymbol{e}(\theta_i + \frac{\pi}{2})) = \frac{1}{3} m_i L_i^2.$$

Finally, we assume that the volumetric force (actually, lineic in our slender case) can be decomposed into the following contributions:

- The gravity, which we will assume along the y axis. $\boldsymbol{g}_i^g = -\rho_i r_i g \boldsymbol{e}_y$, and $\boldsymbol{l}_i^g = \boldsymbol{0}$.
- A viscous air friction term opposing the velocity at each point of the rod, $\mathbf{g}^a(\boldsymbol{x}) = -\eta\mathbf{v}(\boldsymbol{x})$,

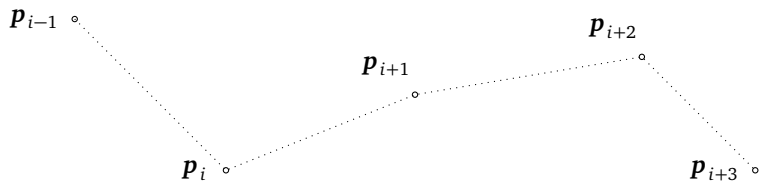


Figure 2.3: *N-tuple pendulum modeled with point passes. Degrees of freedom are limited to the positions ($\boldsymbol{p}_i$), and each element's length is maintained by an holonomic constraint $\|\boldsymbol{p}_{i+1} - \boldsymbol{p}_i| = 2L_i$.*

yielding

$$\begin{aligned}
\boldsymbol{g}_i^a &= -\eta \int_{s=-L_i}^{L_i} \mathbf{v}_i(\boldsymbol{x})\mathrm{d}s = 2L_i\eta\boldsymbol{v}_i \\
\boldsymbol{l}_i^a &= -\eta \int_{s=-L_i}^{L_i} (\boldsymbol{x}(s) - \boldsymbol{p}_i) \wedge \mathbf{v}_i(\boldsymbol{x}(s))\mathrm{d}s \\
&= -\eta \int_{s=-L_i}^{L_i} s\boldsymbol{e}(\theta_i) \wedge (\omega_i(s\boldsymbol{e}(\theta_i)))\,\mathrm{d}s \\
&= -\frac{2}{3}L_i^3\omega_i.
\end{aligned}$$

- The force derived from a bending energy $E_{i,j}^b$, which, assuming a Hookean torsion spring, is quadratic in the bending angle $\alpha_{i,j} := \theta_j - \theta_i$ between adjacent rods: $E_{i,j}^b := \frac{1}{2}K\alpha_{i,j}^2$. The bending energy $E_{i,j}^b$ therefore causes the j^{th} element to induce a torque $K\alpha_{i,j}$ on the i^{th} element. Writing the total bending force and torque applied by the i^{th} element at the link between the i^{th} and $(i+1)^{\text{th}}$ elements gives

$$\begin{aligned}
\mathbf{0} &= \int_{-L}^{L} \mathbf{g}_{i\to i+1}^b(s)\mathrm{d}s \\
-K\alpha_{i,i+1} &= \int_{-L}^{L} (s-L)\boldsymbol{e}(\theta_i) \wedge \mathbf{g}_{i\to i+1}^b(s)\mathrm{d}s = \int_{-L}^{L} s\boldsymbol{e}(\theta_i) \wedge \mathbf{g}_{i\to i+1}^b(s)\mathrm{d}s,
\end{aligned}$$

from which we deduce $\boldsymbol{g}_i^b = 0$ and

$$\boldsymbol{l}_i^b = -K\left(\alpha_{(i-1),i} - \alpha_{i,(i+1)}\right) = -K\left(2\theta_i - \theta_{i-1} - \theta_{i+1}\right).$$

To summarize, the dynamics of our articulated chain are governed by a second-order ordinary differential equation with a banded, symmetric positive semi-definite mass-matrix, non-linear in $\boldsymbol{Q}$. The force $\boldsymbol{f}$ is linear in $\boldsymbol{Q}$ and $\dot{\boldsymbol{Q}}$, and the constraints are non-linear in $\boldsymbol{Q}$.

2.1.2 Lumped system

Another popular way of modeling deformable objects is to consider them as a set of point at which all of the body's mass is concentrated, possibly with constraints on their relative positions. That is, writing $\rho_i(\boldsymbol{x}) = \delta(\boldsymbol{x} - \boldsymbol{p}_i)$, where δ denotes the Dirac distribution such that for any field $\mathbf{f}$ over Ω, $\int_\Omega \mathbf{f}(\boldsymbol{x})\delta(\boldsymbol{y})\mathrm{d}\boldsymbol{x} = \mathbf{f}(\boldsymbol{y})$. We see immediately that with such a mass distribution, $I_i = \mathbf{0}$ and $\boldsymbol{l}_i = 0$; therefore, we need only concern ourselves with the evolution of the position of linear velocity variables $\boldsymbol{p}_i$ and $\boldsymbol{v}_i$, and can discard $\boldsymbol{q}_i$ and $\boldsymbol{\omega}_i$. The mass matrix M is then constant, diagonal and positive definite, simplifying greatly the numerical integration of the dynamical

system. However, this approach is not without drawbacks; the discretization of the physical model might be less accurate, and the forces and constraints will often be nonlinear.

A N-tuple pendulum can be modeled in this framework by positioning the point masses at the links between consecutive elements, as well as at the start and end points of the chain (Figure 2.3). For our chain of N elements, we therefore have $(N+1)$ point masses with coordinates $(\boldsymbol{p}_i)$, with the constraint that for $1 \leq i \leq 0$, $\|\boldsymbol{p}_{i+1} - \boldsymbol{p}_i\| = L_i$. The mass associated to the i^{th} node is then $m_i := \rho(r_{i-1}L_{i-1} + r_i L_i)$ (with the convention $L_{-1} = 0$). Note that this model is *not* equivalent to the chain of rigid-bodies presented in the previous section; the kinetic energy of an individual slender rod is $\frac{m}{2}\left(\boldsymbol{v}^2 + \frac{1}{3}L^2\omega_i^2\right)$, while that of the corresponding two-masses system would be $\frac{m}{2}\left(\boldsymbol{v}^2 + L^2\omega_i^2\right)$. Our goal here is simply to give an insight about the structure of the resulting equations; for the rigid-body chain version of the N-tuple pendulum, only the value (but not the structure) of the inertia matrix and air friction torque would have to be modified.

Now, the forces acting on the node $\boldsymbol{p}_i$ are the sum of:

1. the gravity contribution, $\boldsymbol{f}_i^g = -m_i g \boldsymbol{e}_y$.
2. the force induced by air friction, $\boldsymbol{f}_i^a = -(L_{i-1} + L_i)\eta \boldsymbol{v}_i$.
3. the bending force derived from the bending energy $E_{i,j}^b = \frac{1}{2}K\theta_{i,j}^2$,

$$\boldsymbol{f}_i^b = -\alpha_{(i-2),(i-1)}^{\intercal}\frac{\partial \alpha_{(i-2),(i-1)}}{\partial \boldsymbol{p}_i} - \alpha_{(i-1),i}^{\intercal}\frac{\partial \alpha_{(i-1),i}}{\partial \boldsymbol{p}_i} - \alpha_{i,(i+1)}^{\intercal}\frac{\partial \alpha_{i,(i+1)}}{\partial \boldsymbol{p}_i}$$
$$\alpha_{i,i+1} = \sin^{-1}\left(\frac{\|\boldsymbol{y}_{i,i+1}\|}{4L_{i-1}L_i}\right)\frac{\boldsymbol{y}_{i,i+1}}{\|\boldsymbol{y}_{i,i+1}\|}$$
$$\boldsymbol{y}_{i,j} := (\boldsymbol{p}_i - \boldsymbol{p}_{i-1}) \wedge (\boldsymbol{p}_{i+1} - \boldsymbol{p}_i),$$

which is nonlinear in $\boldsymbol{Q}$.

In brief, the lumped system boasts a diagonal, positive-definite mass matrix, but suffers from highly nonlinear constraints and forces.

2.1.3 Lagrangian mechanics

In the two previous approaches, modeling any complex system led to the introduction of constraints, which might seriously complicate the process of solving the dynamics differential equation. In contrast, Lagrangian mechanics opens the way for using alternative sets of coordinates. In particular, one may choose to directly parameterize the manifold of admissible kinematic variables with so-called *reduced* coordinates, which will eventually lead to a constraint-free system. For the sake of simplicity, we only consider this case below.

We consider a non-conservative dynamical system described by a set of $m \in \mathbb{N}$ kinematic variables $\boldsymbol{q}$ and its time-derivative $\boldsymbol{v} := \frac{\mathrm{d}\boldsymbol{q}}{\mathrm{d}t}$, the so-called generalized velocity. Assuming that $\boldsymbol{q}$ are reduced coordinates, any infinitesimal change $\delta\boldsymbol{q}$ yields a *virtual* displacement $\delta\mathbf{x} = \sum \frac{\partial \mathbf{x}}{\partial \boldsymbol{q}_i}\delta\boldsymbol{q}_i$. Expressing Newtons' second law of motion for any of those virtual displacements gives

$$\int_{s\in\Omega}\left(\frac{\partial \mathbf{x}}{\partial \boldsymbol{q}}(t,\boldsymbol{s},\boldsymbol{q})\right)^{\intercal}\rho(\boldsymbol{s})\frac{\mathrm{d}\mathbf{v}}{\mathrm{d}t}(t,\boldsymbol{s},\boldsymbol{q},\boldsymbol{v})\mathrm{d}\boldsymbol{s} = \int_{s\in\Omega}\left(\frac{\partial \mathbf{x}}{\partial \boldsymbol{q}}(t,\boldsymbol{s},\boldsymbol{q})\right)^{\intercal}\mathbf{g}(t,\mathbf{x}(t,\boldsymbol{s},\boldsymbol{q}),\mathbf{v}(t,\boldsymbol{s},\boldsymbol{q},\boldsymbol{v}))\mathrm{d}\boldsymbol{s}$$

where $\mathbf{g}$ is the external force applied at each point of Ω. Moreover, we have also

$$\begin{aligned}
&\int_{s\in\Omega}\rho(\boldsymbol{s})\frac{\mathrm{d}\mathbf{v}}{\mathrm{d}t}(t,\boldsymbol{s},\boldsymbol{q},\boldsymbol{v})\frac{\partial \mathbf{x}}{\partial \boldsymbol{q}}(t,\boldsymbol{s},\boldsymbol{q})\mathrm{d}\boldsymbol{s}\\
&= \int_{s\in\Omega}\rho(\boldsymbol{s})\left[\frac{\mathrm{d}}{\mathrm{d}t}\left(\mathbf{v}(t,\boldsymbol{s},\boldsymbol{q},\boldsymbol{v})\frac{\partial \mathbf{x}}{\partial \boldsymbol{q}}(t,\boldsymbol{s},\boldsymbol{q})\right) - \mathbf{v}(t,\boldsymbol{s},\boldsymbol{q},\boldsymbol{v})\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial \mathbf{x}}{\partial \boldsymbol{q}}(t,\boldsymbol{s},\boldsymbol{q})\right]\mathrm{d}\boldsymbol{s}\\
&= \int_{s\in\Omega}\rho(\boldsymbol{s})\left[\frac{\mathrm{d}}{\mathrm{d}t}\left(\mathbf{v}(t,\boldsymbol{s},\boldsymbol{q},\boldsymbol{v})\frac{\partial \mathbf{v}}{\partial \boldsymbol{v}}(t,\boldsymbol{s},\boldsymbol{q},\boldsymbol{v})\right) - \mathbf{v}(t,\boldsymbol{s},\boldsymbol{q},\boldsymbol{v})\frac{\partial \mathbf{v}}{\partial \boldsymbol{q}}(t,\boldsymbol{s},\boldsymbol{q},\boldsymbol{v})\right]\mathrm{d}\boldsymbol{s}\\
&= \frac{1}{2}\int_{s\in\Omega}\rho(\boldsymbol{s})\left[\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial \mathbf{v}^2}{\partial \boldsymbol{v}}(t,\boldsymbol{s},\boldsymbol{q},\boldsymbol{v}) - \frac{\partial \mathbf{v}^2}{\partial \boldsymbol{q}}(t,\boldsymbol{s},\boldsymbol{q},\boldsymbol{v})\right]\mathrm{d}\boldsymbol{s}.
\end{aligned}$$

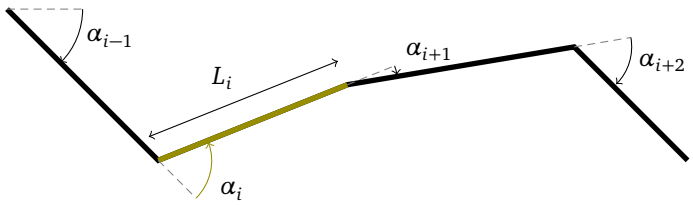


Figure 2.4: *With the Lagrangian approach and reduced coordinates, the angles* (α_i) *between successive elements are the sole degrees of freedom for our 2D articulated chain, and no supplemental constraint is necessary. However, the position of each segment depends on the angle at* all *prior joints, yielding a* dense *numerical system.*

Introducing the kinetic energy $\mathscr{E}^c$ of the system,

$$\mathscr{E}^c(t,\boldsymbol{q},\boldsymbol{v}) := \frac{1}{2}\int_{s\in\Omega} \rho(\boldsymbol{s})\mathbf{v}^2(t,\boldsymbol{s},\boldsymbol{q},\boldsymbol{v})\mathrm{d}\boldsymbol{s},$$

we obtain the Euler–Lagrange dynamics equation[2],

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial \mathscr{E}^c}{\partial \boldsymbol{v}}(t,\boldsymbol{q},\boldsymbol{v}) - \frac{\partial \mathscr{E}^c}{\partial \boldsymbol{q}}(t,\boldsymbol{q},\boldsymbol{v}) = \int_{s\in\Omega}\left(\frac{\partial \mathbf{x}}{\partial \boldsymbol{q}}(t,\boldsymbol{s},\boldsymbol{q})\right)^{\intercal}\mathbf{g}(t,\boldsymbol{s},\boldsymbol{q},\boldsymbol{v})\mathrm{d}\boldsymbol{s}. \tag{2.7}$$

It is convenient to express forces that can be derived from potentials as such; for instance, a large class of systems can be written as

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial \mathscr{E}^c}{\partial \boldsymbol{v}}(t,\boldsymbol{q},\boldsymbol{v}) - \frac{\partial \mathscr{E}^c}{\partial \boldsymbol{q}}(t,\boldsymbol{q},\boldsymbol{v}) + \frac{\partial \mathscr{E}^p}{\partial \boldsymbol{q}}(\boldsymbol{q}) = -\frac{\partial \mathscr{D}}{\partial \boldsymbol{v}}(\boldsymbol{q},\boldsymbol{v}) \tag{2.8}$$

where $\mathscr{E}^p$ is the potential energy of the system and $\mathscr{D}$ is a dissipation potential.

$\mathscr{E}^c$ being generally quadratic w.r.t. $\boldsymbol{v}$, expressing the derivatives and integral of Equation (2.7) leads to

$$M(t,\boldsymbol{q})\frac{\mathrm{d}\boldsymbol{v}}{\mathrm{d}t} = \boldsymbol{f}(t,\boldsymbol{q},\boldsymbol{v})$$

where $M(t,\boldsymbol{q})$ is symmetric positive semi-definite, with $M_{ij} = \int_\Omega \rho(\boldsymbol{s})\frac{\partial^2 \mathbf{v}}{\partial \boldsymbol{v}_i \partial \boldsymbol{v}_j}(t,\boldsymbol{s},\boldsymbol{q},\boldsymbol{v})\mathrm{d}\boldsymbol{s}$, and the generalized force $\boldsymbol{f}$ groups terms from internal, external and inertial forces. The expression of the dynamics of the system in the reduced coordinates $\boldsymbol{q}$ is therefore similar to that of the unconstrained dynamics equation (2.5) in the framework of Section 2.1.1. However, as we will now see on our 2D articulated chain example, the mass-matrix $M(t,\boldsymbol{q})$ will not necessarily be sparse anymore.

Articulated chain Let $\boldsymbol{q} = (\alpha_i) \in \mathbb{R}^N$, with α_i the angle between the $(i-1)^{\text{th}}$ and the i^{th} segments, and with α_0 defined w.r.t. to the clamping angle θ_0 which we will assume to be 0. The generalized velocity is $\boldsymbol{v} = (\omega_i)$, with $\omega_i = \frac{\mathrm{d}\alpha_i}{\mathrm{d}t}$.

Potential energy The potential energy is the sum of two terms, the bending energy E^b and the contribution of gravity E^g. We have $E^b_{i-1,i}(\boldsymbol{q}) = \frac{1}{2}K\alpha_i^2$, and therefore

$$\frac{\partial E^b}{\partial \boldsymbol{q}}(\boldsymbol{q}) = \sum_i \frac{\partial E^b_{i-1,i}}{\partial \boldsymbol{q}}(\boldsymbol{q}) = K\boldsymbol{q}^{\intercal}.$$

[2]First established by Joseph Louis Lagrange from the principle of least action in 1756, then from D'Alembert principle in 1788.

The potential energy associated to the gravity is more complex. The position of the point at abscissa s of the i^{th} element can be recovered from a recursive formula $\mathbf{x}_i(s) = \mathbf{x}_{i-1}(L_{i-1}) + (s + L_i)(\Pi_{j=0}^{i} R(\alpha_j))\boldsymbol{e}_x$. and the clamping condition $\mathbf{x}_0 = \mathbf{0}$. We can therefore evaluate

$$E^g(\boldsymbol{q}) = \sum_i \rho_i r_i \int_{-L_i}^{L_i} \langle \mathbf{x}_i(s), \boldsymbol{e}_y \rangle \, \mathrm{d}s = \sum_i m_i \langle \mathbf{x}_i(0), \boldsymbol{e}_y \rangle,$$

and the gravity generalized force, $-\frac{\partial E^g}{\partial \boldsymbol{q}}(\boldsymbol{q})$, can then be computed using

$$\frac{\partial \mathbf{x}_i(s)}{\partial \boldsymbol{q}_j}(\boldsymbol{q}) = \begin{cases} \left(\Pi_{k=1}^{j-1} R(\alpha_k)\right) R(\alpha_j + \frac{\pi}{2}) \left(\mathbf{x}_i(s) - \mathbf{x}_j(-L_j)\right) & \text{if } j \leq i \\ \mathbf{0} & \text{if } j > i \end{cases}$$

Air friction The generalized force $\boldsymbol{f}^a$ due to air friction derives from the dissipation potential

$$\mathscr{D}(\boldsymbol{q}, \boldsymbol{v}) := \frac{\eta}{2} \sum_i \int_{-L_i}^{L_i} \mathbf{v}_i^2(s) \mathrm{d}s,$$

where $\mathbf{v}_i(s)$ can be once again be computed using a recursive formula,

$$\mathbf{v}_i(s) = \mathbf{v}_{i-1}(L) + (s + L) \left(\sum_{j=1}^{i} \omega_i \right) \boldsymbol{e} \left(\frac{\pi}{2} + \sum_{j=1}^{i} \alpha_j \right). \tag{2.9}$$

The expression the friction force is then

$$\boldsymbol{f}_a = -\frac{\partial \mathscr{D}(\boldsymbol{q}, \boldsymbol{v})}{\partial \boldsymbol{v}} = -\eta \sum_i \int_{-L_i}^{L_i} \left(\frac{\partial \mathbf{x}_i}{\partial \boldsymbol{q}}(s) \right)^{\intercal} \mathbf{v}_i(s) \mathrm{d}s,$$

Inertia Finally, it remains to consider the derivatives of the kinetic energy $\mathscr{E}^c$ w.r.t. the generalized positions and velocity. We have

$$\mathscr{E}^c = \frac{1}{2} \sum_i \int_{-L_i}^{L_i} \mathbf{v}_i^2(s) \mathrm{d}s = \frac{1}{2} \sum_i m_i \left(\mathbf{v}_i^2(0) + \frac{1}{3} L_i^2 \omega_i^2 \right).$$

We see from Equation (2.9) that $\mathbf{v}_N(0)$ depends linearly on all the velocity components (ω_i); indeed,

$$\frac{\partial \mathbf{v}_N(0)}{\partial \omega_i} = (s + L)\boldsymbol{e} \left(\frac{\pi}{2} + \sum_{j=1}^{N} \alpha_j \right) + 2L \sum_{k=i}^{N-1} \boldsymbol{e} \left(\frac{\pi}{2} + \sum_{j=1}^{k} \alpha_j \right).$$

Therefore, $\forall i, j \in [1, N]^2$,

$$\frac{\partial^2 \mathbf{v}_N^2(0)}{\partial \omega_i \partial \omega_j} = \left(\frac{\partial \mathbf{v}_N(0)}{\partial \omega_i} \right) \left(\frac{\partial \mathbf{v}_N(0)}{\partial \omega_j} \right)^{\intercal}$$

and we deduce that the mass matrix $M(t, \boldsymbol{q})$ is generally dense.

2.1.4 Discussion

While being far from exhaustive, we illustrated a few different approaches for deriving the equations of motion of a mechanical system with kinematic constraints. The logical follow-up question is then: which one should we use ? There is no definitive answer, but we can compare the different approaches on selected criterions.

To this aim, let us remark that all previous derivations can be coalesced into a single canonical set of equations,

$$M(t,\boldsymbol{q})\frac{\mathrm{d}\boldsymbol{v}}{\mathrm{d}t} = \boldsymbol{f}(t,\boldsymbol{q},\boldsymbol{v}) + \left(\frac{\partial C}{\partial \boldsymbol{q}}(t,\boldsymbol{q})\right)^{\intercal}\boldsymbol{\lambda}$$
$$C(t,\boldsymbol{q}) = \mathbf{0}$$

where $\boldsymbol{q}$ is a set a generalized coordinates, and $\boldsymbol{v}$ a set of generalized velocities. However, the structure of the mass-matrix M, the properties of force $\boldsymbol{f}$, and the kinematic constraints C showed substantial variations depending on the nature of coordinates.

Modeling capacity Reduced coordinates may not always be adequate. For once, not every system can be described with solely holonomic constraints, and even then, finding a constraint-free parametrization is a hard problem — such is the case of inextensible shells (Casati 2015). Note, however that the Lagrangian approach does not prescribe the use of reduced coordinates; it is entirely possible, using again D'Alembert's principle, to add supplemental constraints to the Lagrange equation of motion (2.7), yielding an "intermediate" set of coordinates. However, the Lagrange equation (2.7) does require that the velocity variable $\boldsymbol{v}$ be the derivative in time of the position variable. For instance, this prohibits using a quaternion for the position variable and an angular velocity as the velocity variable, like we did for rigid bodies.

Conversely, point masses are a very rough discretization of a mechanical system, and might not model correctly some physical quantities, such as the angular momentum in our 2D example. Getting more faithful to the continuous model then means adding more point masses and constraints, making the equations more expensive to solve numerically.

Numerical integration In the presence of holonomic constraints, the equations of motion boil down to an index-3 differential algebraic equation (DAE). Numerical integration of such equations have been extensively studied in the literature, we point the reader to (Haddouni 2015) for a recent review.

On the one hand, the absence of additional constraints is obviously a huge advantage of the Lagrangian approach with reduced coordinates, as it allows the use of much simpler integrators. However, in practice supplemental constraints may be required to allow artistic control of the simulation, and deriving a new reduced kinematic model for each new scene is not practical — it is safer to devise a numerical integrator that can handle kinematic constraints anyway.

On the other hand, the main advantage of the point-masses method is that the mass matrix will always be diagonal and positive-definite, and thus trivial to inverse; on the contrary, slender 3D structures may have vanishing inertia terms when modeled with the Lagrangian of articulated rigid-body approach. Moreover, the mass matrix of reduced-coordinates models can be dense, leading to more expensive computations.

Another important point deals with the structure of the internal forces; having them depend linearly on the kinematic variables may greatly simplify the numerical integration process, as we shall see later. In our 2D example, we saw that all three forces were linear for the chain of rigid-bodies, while the bending force was nonlinear of the point-masses discretization, and both gravity and air friction forces were nonlinear for the reduced coordinates model. One may thus choose one or the other model depending on the respective influence of each force term.

2.2 Contacts

For the sake of simplicity, we first consider a system without kinematic constraints. Using any kind of coordinates, the equation of motion may be written as

$$M(t,\boldsymbol{q})\frac{\mathrm{d}\boldsymbol{v}}{\mathrm{d}t} = \boldsymbol{f}(t,\boldsymbol{q},\boldsymbol{v}) + \boldsymbol{f}^c, \quad \boldsymbol{f}^c := \int_{s\in\Omega}\left(\frac{\partial \mathbf{v}}{\partial \boldsymbol{v}}(t,\boldsymbol{s},\boldsymbol{q})\right)^{\intercal}\mathbf{g}^c(\boldsymbol{s})\mathrm{d}\boldsymbol{s},$$

where $\mathbf{g}^c$ is a supplemental force density due to the contacts.

2.2.1 Continuous-time equations of motion with contacts

First, let us discretize our contact surface as a finite number n of contact points $(\boldsymbol{c}_i)$, at which the contact normal is well-defined. Generating this set is a non trivial task, and the chosen process will largely influence the numerical resolution of the system's dynamics; however, we consider it as out of the scope of this thesis. Let $\boldsymbol{u}_i$ denote the relative velocity at the i^{th} contact point. If the i^{th} contact is between two points of our mechanical system (*self* contact), then there exist $\boldsymbol{s}_{i,1}$ and $\boldsymbol{s}_{i,2}$ such that $\boldsymbol{u}_i(t,\boldsymbol{q},\boldsymbol{v}) = \mathbf{v}(t,\boldsymbol{s}_{i,1},\boldsymbol{q},\boldsymbol{v}) - \mathbf{v}(t,\boldsymbol{s}_{i,2},\boldsymbol{q},\boldsymbol{v})$. Otherwise, the contact is between a point of our system and an external object (*external* contact), and $\boldsymbol{u}_i(t,\boldsymbol{q},\boldsymbol{v}) = \mathbf{v}(t,\boldsymbol{s}_{i,1},\boldsymbol{q},\boldsymbol{v}) - \boldsymbol{w}^{\text{ext}}(t)$, where $\boldsymbol{w}^{\text{ext}}(t)$ is the velocity of the external object at $\boldsymbol{c}_i$. Let $\boldsymbol{u} = (\boldsymbol{u}_i)$.

The contact force density $\boldsymbol{g}^c$ can be then be expressed as a sum of n punctual forces $(\boldsymbol{r}_i)$, $\boldsymbol{g}^c = \sum_i \boldsymbol{r}_i \delta(\boldsymbol{x} - \boldsymbol{c}_i)$. We get

$$\begin{aligned}\boldsymbol{f}^c &= \sum_{i=1}^{n} \begin{cases} \left(\frac{\partial \mathbf{v}}{\partial \boldsymbol{v}}(t,\boldsymbol{s}_{i,1},\boldsymbol{q})\right)^{\intercal} \boldsymbol{r}_i & \text{if } i \text{ is a self contact} \\ \left(\frac{\partial \mathbf{v}}{\partial \boldsymbol{v}}(t,\boldsymbol{s}_{i,1},\boldsymbol{q}) - \frac{\mathbf{v}}{\boldsymbol{v}}(t,\boldsymbol{s}_{i,2},\boldsymbol{q})\right)^{\intercal} \boldsymbol{r}_i & \text{if } i \text{ is an external contact} \end{cases} \\ &= \left(\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{v}}(t,\boldsymbol{q})\right)^{\intercal} \boldsymbol{r}.\end{aligned}$$

We will assume that $\boldsymbol{u}$ is an affine function of $\boldsymbol{v}$; that is, $\boldsymbol{u}(t,\boldsymbol{q}) := H(t,\boldsymbol{q})\boldsymbol{v} + \boldsymbol{w}(t)$ with $H(t,\boldsymbol{q}) = \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{v}}(t,\boldsymbol{q})$. Our equations of motion with unilateral contact can then be summarized as

$$\begin{cases} M(t,\boldsymbol{q})\dfrac{\mathrm{d}\boldsymbol{v}}{\mathrm{d}t} = \boldsymbol{f}(t,\boldsymbol{q},\boldsymbol{v}) + H(t,\boldsymbol{q})^{\intercal}\boldsymbol{r} \\ \qquad \boldsymbol{u} = H(t,\boldsymbol{q})\boldsymbol{v} + \boldsymbol{w}(t) \\ \quad \boldsymbol{u}_i, \boldsymbol{r}_i \text{ satisfy the contact law for } 1 \le i \le n. \end{cases} \tag{2.10}$$

First, we can treat the case of frictionless contacts, or more generally, an associated contact law with the convex cone K as the set of admissible forces. The contact law is then $K^\star \ni \boldsymbol{u}_i \perp \boldsymbol{r}_i \in K$, and we can rewrite system (2.10) as the differential inclusion

$$M(t,\boldsymbol{q})\frac{\mathrm{d}\boldsymbol{v}}{\mathrm{d}t} - \boldsymbol{f}(t,\boldsymbol{q},\boldsymbol{v}) \in -H(t,\boldsymbol{q})^{\intercal}\mathscr{N}_{K^\star}(H_i(\boldsymbol{q})\boldsymbol{u}_i + \boldsymbol{w}_i(t)).$$

Under a regularity assumption, for instance, $\exists \boldsymbol{v}$ such that $\boldsymbol{u}(t,\boldsymbol{q},\boldsymbol{v}) \in \mathrm{int}(K^\star)^n$, we can use Property A.12 on the subdifferential of the precomposition with an affine map to write equivalently our dynamics equation with unilateral contact as

$$M(t,\boldsymbol{q})\frac{\mathrm{d}\boldsymbol{v}}{\mathrm{d}t} - \boldsymbol{f}(t,\boldsymbol{q},\boldsymbol{v}) \in -\mathscr{N}_{C(\boldsymbol{q},t)}(\boldsymbol{v}), \tag{2.11}$$

with the admissible velocity set $V(\boldsymbol{q},t) := \{\boldsymbol{v}, H(t,\boldsymbol{q})\boldsymbol{v} + \boldsymbol{w}(t) \in (K^\star)^n\}$. The associated contact law can therefore be modeled with the addition of a dissipation potential $\mathscr{D}(t,\boldsymbol{q},\boldsymbol{v}) = \mathscr{I}_{V(\boldsymbol{q},t)}(\boldsymbol{v})$ to the equations of motion.

This form of the equations highlights a difficulty in the definition of $\frac{\mathrm{d}\boldsymbol{v}}{\mathrm{d}t}$: as the right-hand-side of (2.11) may be unbounded and non-differentiable w.r.t. $\boldsymbol{v}$, $\boldsymbol{q}$, or t, $\boldsymbol{v}$ might be discontinuous. This is physically intuitive: considering the inelastic impact of a body falling on the ground at time t, its vertical velocity goes instantaneously from a non-zero value at t_- to zero at t_+. The symbol $\frac{\mathrm{d}\boldsymbol{v}}{\mathrm{d}t}$ should instead be understood as a function defined almost everywhere on a time interval $[T_0, T_1]$, such that

$$\boldsymbol{v}(t) = \boldsymbol{v}(T_0) + \int_{T_0}^{t} \frac{\mathrm{d}\boldsymbol{v}}{\mathrm{d}t}\mathrm{d}t \quad \forall t \in [T_0, T_1].$$

Equation (2.11) can then be cast as a *measure differential inclusion*. We will not worry here about the existence of solutions in time to such inclusions, as we will promptly resort to a time-stepping scheme which alleviates most concerns (see e.g., Moreau 1999).

Going back to the Coulomb frictional contact law, our equations of motions read

$$\begin{cases} M(t,\boldsymbol{q})\dfrac{\mathrm{d}\boldsymbol{v}}{\mathrm{d}t} = \boldsymbol{f}(t,\boldsymbol{q},\boldsymbol{v}) + H(t,\boldsymbol{q})^{\intercal}\boldsymbol{r} \\ \qquad \boldsymbol{u} = H(t,\boldsymbol{q})\boldsymbol{v} + \boldsymbol{w}(t) \\ (\boldsymbol{u}_i,\boldsymbol{r}_i) \in \mathscr{C}_{\mu_i} \qquad\qquad \forall 1 \leq i \leq n, \end{cases} \tag{2.12}$$

where $\frac{\mathrm{d}\boldsymbol{v}}{\mathrm{d}t}$ is still to be understood in the sense mentioned above. The Coulomb law does not derive from a dissipation potential, but the equation of motions can still be written once again as a differential inclusion,

$$M(t,\boldsymbol{q})\frac{\mathrm{d}\boldsymbol{v}}{\mathrm{d}t} - \boldsymbol{f}(t,\boldsymbol{q},\boldsymbol{v}) \in -H(t,\boldsymbol{q})^{\intercal}\mathscr{N}_{\left(\Pi\mathscr{K}_{\frac{1}{\mu_i}}\right)}(\tilde{\boldsymbol{u}}), \tag{2.13}$$

where $\tilde{\boldsymbol{u}} = (\tilde{\boldsymbol{u}}_i) = (\boldsymbol{u}_i + \mu_i \|(\boldsymbol{u}_i)_{\mathbf{T}}\| \boldsymbol{n}_i)$ is the contact-wise de Saxcé change of variable defined in Property 1.4.

2.2.2 Time integration

We now consider the problem of finding solution in time to our equations of motion with unilateral contacts. Several algorithms have been developed; we again refer to (Haddouni 2015, chapter 2) for a complete review on this topic.

We can cite two large class of methods. First, event-driven schemes decouple the nonsmooth events (such as impacts), which happen at a countable set of instants (t_i), from the smooth dynamics which hold in every interval between those instants. The main advantage of this approach is the ability to use arbitrarily-high order integration schemes between the nonsmooth events, reaching tight error tolerances at a reasonable computational cost. However, there are also drawbacks:

- A infinite number of events may occur in a finite time interval; this is known as a *Zeno* phenomenon, and is observed in mechanical systems as simple as rigid ball bouncing on the ground with a restitution coefficient in $]0,1[$ (Brogliato 1999);
- Simultaneous nonsmooth events require special care;
- Numerically, detecting events requires the introduction of several thresholds — saying that for instance, two objects will no longer be considered in contact when the gap between them exceeds a certain value. Tuning this threshold parameters is difficult to automate.

A second kind of integration method is usually preferred in practice for systems in which a large number of events may occur, such as large multi-body systems : *time-stepping* methods, which integrate the equation of motion over a time interval $[t^k, t^{k+1} = t^k + \Delta_t]$ regardless of whether or not nonsmooth events occur during the timestep.

Time-stepping methods suppose prior knowledge of the contacts that are going to be active during the timestep; in practice, this means using proximity-based or continuous-time collision detection methods, which we will discuss in the next section. Then, the Moreau–Jean (1987, Moreau 1988) time-stepping algorithm discretizes the continuous-time measure differential equation (2.13) by enforcing that the end-of-step velocity $\boldsymbol{v}^{k+1}$ — or the corresponding relative velocity, $\boldsymbol{u}^{k+1}$ — satisfies the contact constraints.

Velocity-level constraints Consider the i^{th} contact predicted to happen during a timestep $[t^k, t^k + \Delta_t]$, and let $\boldsymbol{h}_i(t)$ denote the gap function, that is, the difference in position at time t of the two points that are going to come into contact at $t_i^c \in [t^k, t^{k+1}]$, as illustrated in Figure 2.5.

For a large Δ_t, $\boldsymbol{h}_{\text{iN}}(t^k)$ might be far from zero, that is, the two points of the future contact may still be far away from each other. However, the velocity constraint that we formulated in the continuous case reads $\boldsymbol{u}_{\text{iN}}^{k+1} \geq 0$, which would prevent the two points to get any closer.

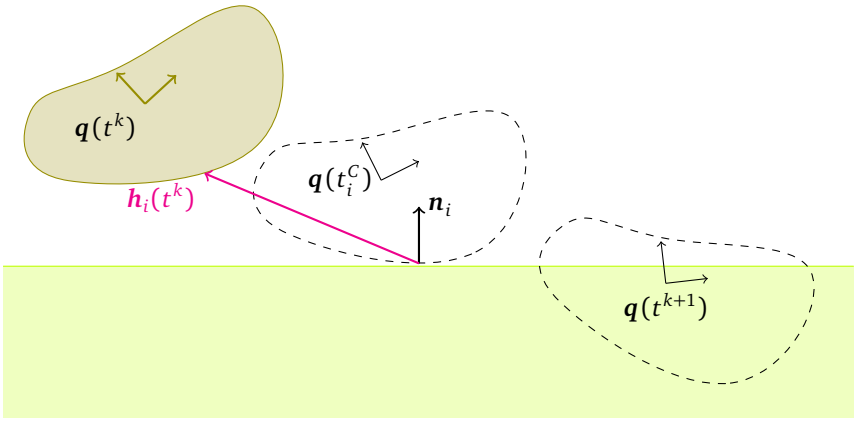


Figure 2.5: *Gap function h_i for a contact between a body with coordinates $q(t)$ and a fixed obstacle predicted to happen at time $t_i^C \in]t^k, t^{k+1}]$.*

Instead, the Moreau-Jean time-stepping algorithm formulates the Signorini constraint as follows:

$$\begin{cases} h_i(t^{k+1})_{\mathrm{iN}} = 0 \text{ and } 0 \le u_{\mathrm{iN}}^{k+1} \perp r_{\mathrm{iN}} \ge 0 \\ \text{or } h_i(t^{k+1})_{\mathrm{iN}} > 0 \text{ and } r_{\mathrm{iN}} = 0, \end{cases}$$

where r is the average of the contact forces over the timestep. In order to avoid introducing a new set of constraints, a modified Signorini condition, $0 \le h_i(t^{k+1}) \perp r_{\mathrm{iN}} \ge 0$, is commonly enforced. Using a first-order integration scheme, it is usual to write $q^{k+1} - q^k = \Delta_t(\alpha v^{k+1} + \alpha v^k)$ with $\alpha \in [0,1]$; $h_i(t^{k+1})$ can then be approximated to the first order as $h_i(t^{k+1}) = h_i(t^k) + \Delta_t u_i^\alpha$, with $u_i^\alpha = \alpha u^{k+1} + (1-\alpha)u^k$. The modified Signorini constraint written at the velocity level then reads

$$0 \le u_{\mathrm{iN}}^{k+1} + \frac{1-\alpha}{\alpha} u_{\mathrm{iN}}^k + \frac{1}{\alpha \Delta_t} h_{\mathrm{iN}}^k \perp r_{\mathrm{iN}} \ge 0. \tag{2.14}$$

This restricts our choice for α; indeed, we can see that for $\alpha < 1$, we may have $\frac{1-\alpha}{\alpha} u_{\mathrm{iN}}^k + \frac{1}{\alpha\Delta_t} h_{\mathrm{iN}}^k < 0$, and therefore the modified Signorini condition (2.14) will not prevent having $u_{\mathrm{iN}}^{k+1} > 0$ simultaneously with $r_{\mathrm{iN}} > 0$. In other terms, we may observe a rebound, which is contrary to our inelastic impact assumption. We will therefore restrict ourselves to $\alpha = 1$, and the end-of-step positions will be given by $q^{k+1} = q^k + \Delta_t v^{k+1}$.

Accounting for the relative motion of the two points before the impact with the modified Signorini condition (2.14) will then just mean adding an offset $\frac{1}{\Delta_t} h(t^k)$ to the affine term $w_i(t)$ of the relative velocity u_i.

Discrete-time equations of motion The Moreau-Jean algorithm approximates the smooth dynamics using a first-order ϑ-scheme, and the acceleration using finite differences. For $\vartheta \in [0,1]$, we note

$$\begin{aligned} t^\vartheta &= \vartheta t^{k+1} + (1-\vartheta) t^k \\ v^\vartheta &= \vartheta v^{k+1} + (1-\vartheta) v^k \\ q^\vartheta &= \vartheta q^{k+1} + (1-\vartheta) q^k \end{aligned}$$

The differential inclusion (2.13) is then integrated over the timestep as [3]

$$M(t^\vartheta, q^\vartheta)(v^{k+1} - v^k) - f(t^\vartheta, q^\vartheta, v^\vartheta) = \Delta_t H(t^\vartheta, q^\vartheta)^\intercal r.$$

[3] As mentioned by Cadoux (2009), f might depend on v and may thus be nonsmooth; the end-of-step velocity may be considered in this case.

Enforcing holonomic constraints to be satisfied at t^ϑ, our discrete equations of motion read

$$\begin{cases} M(t^\vartheta, \boldsymbol{q}^\vartheta)\dfrac{\boldsymbol{v}^{k+1}-\boldsymbol{v}^k}{\Delta_t} = \boldsymbol{f}(t^\vartheta, \boldsymbol{q}^\vartheta, \boldsymbol{v}^\vartheta) + H(t^\vartheta, \boldsymbol{q}^\vartheta)^\intercal \boldsymbol{r} + \left(\dfrac{\partial C}{\partial \boldsymbol{q}}(t^\vartheta, \boldsymbol{q}^\vartheta)\right)^\intercal \boldsymbol{\lambda} \\ C(t^\vartheta, \boldsymbol{q}^\vartheta) = \boldsymbol{0} \\ \boldsymbol{u} = H(t^\vartheta, \boldsymbol{q}^\vartheta)\boldsymbol{v}^{k+1} + \boldsymbol{w}(t^\vartheta) + \dfrac{1}{\Delta_t}\boldsymbol{h}(t^k) \\ (\boldsymbol{u}_i, \boldsymbol{r}_i) \in \mathscr{C}_{\mu_i} \quad \forall 1 \le i \le n. \end{cases} \tag{2.15}$$

Linearized dynamics Solving the implicit nonlinear system (2.15) can then be decomposed as a sequence of problems with explicit mass matrix, internal forces and constraints Jacobians using a quasi-Newton algorithm; see, e.g., (Jean 1999; Kaufman, Tamstorf, et al. 2014).

Indeed, let the sequence $(\boldsymbol{v}^{k,l}, \boldsymbol{q}^{k,l})$ be recursively defined as the solution of the problem (2.16),

$$\begin{cases} \boldsymbol{q}^{k,l+1} = \boldsymbol{q}^k + \Delta_t \boldsymbol{v}^{k,l+1} \\ \tilde{M}^l \dfrac{\boldsymbol{v}^{k,l+1}-\boldsymbol{v}^{k,l}}{\Delta_t} = \boldsymbol{f}(t^\vartheta, \boldsymbol{q}^{\vartheta,l}, \boldsymbol{v}^{\vartheta,l}) - M^l \dfrac{\boldsymbol{v}^{k,l}-\boldsymbol{v}^k}{\Delta_t} + H^{l,\intercal}\boldsymbol{r} + G^{l,\intercal}\boldsymbol{\lambda} \\ \tilde{M}^l := M^l - \vartheta\Delta_t \dfrac{\partial \boldsymbol{f}}{\partial \boldsymbol{v}}(t^\vartheta, \boldsymbol{q}^{\vartheta,l}, \boldsymbol{v}^{\vartheta,l}) - \vartheta\Delta_t^2 \dfrac{\partial \boldsymbol{f}}{\partial \boldsymbol{q}}(t^\vartheta, \boldsymbol{q}^{\vartheta,l}, \boldsymbol{v}^{\vartheta,l}) \\ G^l \boldsymbol{v}^{k,l+1} = G^l \boldsymbol{v}^{k,l} - C(\boldsymbol{q}^{\vartheta,l}) \\ \boldsymbol{u} = H^l \boldsymbol{v}^{k,l+1} + \boldsymbol{w}(t^\vartheta) + \dfrac{1}{\Delta_t}\boldsymbol{h}(t^k) \\ (\boldsymbol{u}_i, \boldsymbol{r}_i) \in \mathscr{C}_{\mu_i} \quad \forall 1 \le i \le n, \end{cases} \tag{2.16}$$

with $M^l := M(t^\vartheta, \boldsymbol{q}^{\vartheta,l})$, $H^l := H(t^\vartheta, \boldsymbol{q}^{\vartheta,l})$, $G^l := \partial C \boldsymbol{q}(t^\vartheta, \boldsymbol{q}^{\vartheta,l})$, $\boldsymbol{q}^{\vartheta,l} := \vartheta \boldsymbol{q}^{k,l} + (1-\theta)\boldsymbol{q}^k$, $\boldsymbol{v}^{\vartheta,l} := \vartheta \boldsymbol{v}^{k,l} + (1-\theta)\boldsymbol{v}^k$, and $(\boldsymbol{v}^{k,0}, \boldsymbol{q}^{k,0}) = (\boldsymbol{v}^k, \boldsymbol{q}^k)$. Then any value towards which this sequence converges will be a solution of the equations of motion (2.15). If $\boldsymbol{f}$ derives from potentials, then $\tilde{M}$ should remain symmetric; however $\tilde{M}$ may non-remain positive semi-definite for large values of Δ_t. Other approximations should then be used to estimate the Hessian, for instance the Gauss–Newton method for quadratic potentials.

In practice, this algorithm can be costly; even if the solution at iteration l is a good initial guess for the solution at $l+1$, solving each subproblem may require expensive preprocessing steps. The algorithm is thus (2.16) often truncated to its first step, yielding the so-called *linearized* equations of motion. We will actually adopt this strategy for all of our applications. However, in the presence of highly nonlinear forces, this might worsen the stability or energy conservation properties of the time-stepping scheme (Kaufman, Tamstorf, et al. 2014).

Limitations The time-stepping approach also suffers from limitations. First, it is a low-order method, and thus requires small timesteps to approximate the continuous solution, even when no nonsmooth events occur. Note that Acary (2009) proposed a first method for improving the order of accuracy of the Moreau time-stepping scheme, which performed well on academic examples.

Then, in the above formulation, the contact constraints are formulated on the velocity, and may be subject to numerical drift. If $\boldsymbol{h}_{\text{iN}}(t)$ becomes negative, then once again we may have $\boldsymbol{u}_{\text{iN}}^{k+1}\boldsymbol{r}_{\text{iN}} > 0$, and observe a rebound. On the other hand, if we clamp $\boldsymbol{h}_{\text{iN}}(t)$ to positive values, the unilateral contact constraints may be violated.

Finally, as already mentioned, an important limitation of the time-stepping algorithm is that all contacts that may occur during the timestep must be predicted, which is hard problem, as we are going to see in the next section.

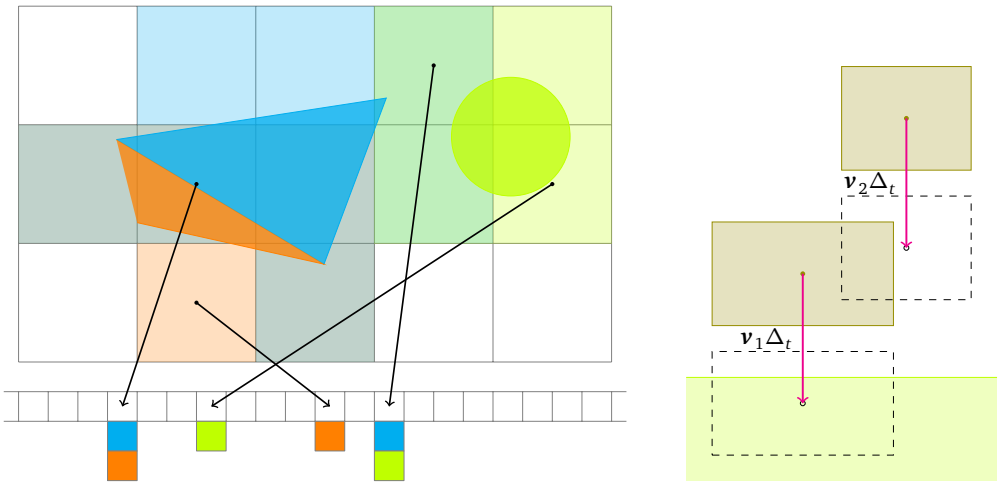


Figure 2.6: ***(a)*** *Spatial hashing of two objects. Only simplices located in cells containing more than one object will have to be checked for collisions during the fine pass.* ***(b)*** *Failure case for continuous-time collision detection: the contact between the two falling bodies can only be detected once the contact between the green and the middle body has been resolved.*

2.2.3 Collision detection

In this section, we briefly discuss a few concepts related to the detection of possible collisions within one timestep. For both a theoretical overview and practical implementation of fundamental algorithms, we refer the reader to Ericson (2004).

On the one hand, or slow enough relative velocities, or small enough timesteps, a very simple and efficient way of performing collision detection is to use a proximity criterion; two points will be considered in contact if the distance between them is below a detection radius r. As it would not be computationally tractable to compute the distance between every pair of points of the system, several approaches have been developed to speed-up the computation. Generally, they amount to a coarse pass, yielding a conservative list of possible collisions, then a finer pass checking the real distance between all the possible pairs. One such simple yet effective acceleration structure is the spatial hashing of Teschner et al. (2003). First, each simulated object is discretized as a set of simplices (S_i), such as balls, segments or triangles. The space $\mathbb{R}^3$ is then partitioned as a regular grid, and to each grid cell $\boldsymbol{c}$ is associated an index $\text{hash}(\boldsymbol{c}) \in \mathbb{N}$. This mapping is used to define a hash-map $\mathscr{H}$ associating to each cell the set of simplices that are within a radius r of this cell: $\mathscr{H}[\text{hash}(\boldsymbol{c})] = \{i \in \mathbb{N}, d(S_i, \boldsymbol{c}) < r\}$. The fine pass simply amounts to checking for collisions between the simplices of each of the non-empty hash-map cells; that is, computing $d(S_i, S_j)\ \forall (i,j) \in \mathscr{H}(k), i \neq j,\ \forall k \in \mathbb{N}$, and for each pair such that $d(S_i, S_j) < r$, computing the closest point on each simplex and deducing the collision normal. Compared to representing the grid as a dense array, this approach has the advantage of a much lower memory footprint when the simulated objects occupy only sparse portions of the domain. However, the grid cell size has to be chosen carefully; if it is too coarse or too fine, the number of distance computations will increase.

On the other hand, for fast-moving objects or large timesteps, proximity-based collision detection will miss some collisions. In this case, the preferred approach is continuous-time collision detection, i.e., checking for collisions all along the trajectories of the simulated objects; obviously, this will require more complex and expensive methods. However, for several pairs of simple geometric shapes, collisions may be found by looking at the roots of a low-order polynomial. For instance, collisions between triangular meshes may be of two kinds: vertex–face collisions and

edge–edge collisions. Assuming a constant rigid motion for each triangle over the timestep, both correspond to the roots of a cubic polynomial. Brochu et al. (2012) proposed a geometrically-exact detection method for such intersections, avoiding the rounding errors associated with the polynomial root-finding approach.

Yet, in order to compute the collisions along a trajectory, we first need to know the trajectory. Which we don't, since the trajectory will be influenced by the contact forces, and obviously, to compute the contact forces, we first need to know the contact points. We therefore need to compromise, and will use unconstrained velocity (the solution of the unconstrained equations of motion) to define the trajectory that will be used for continuous-time collision detection. However, some contacts may not be predicted before the resolution of other collisions. For instance, consider two thin bodies A and B, with B above A, falling on the ground G at the same speed. If Δ_t is big enough, they might impact the ground during the same time step; the continuous-time collision detection process will then predict the contacts A/G and B/G, but not the contact A/B, and the two bodies may end up in the same place. Such situations are likely to happen in the simulation of layered cloth, and may require sophisticated heuristics.

Another solution, if the computational cost is deemed acceptable, would be to perform a collision detection step at each iteration of the algorithm (2.16).

2.3 Discrete Coulomb Friction Problem

Let us refer to the system of equations that we have to solve at each step of the algorithm (2.16) as a *Discrete Coulomb Friction Problem* (DCFP), and let us write it using a lighter notation as

$$\begin{cases} M\boldsymbol{v} = \boldsymbol{f} + C^\intercal\boldsymbol{\lambda} + H^\intercal\boldsymbol{r} \\ C\boldsymbol{v} = \boldsymbol{k} \\ \boldsymbol{u} = H\boldsymbol{v} + \boldsymbol{w} \\ (\boldsymbol{u}_i, \boldsymbol{r}_i) \in \mathscr{C}_{\mu_i} \qquad\qquad \forall 1 \le i \le n, \end{cases} \tag{2.17}$$

where the unknowns are $\boldsymbol{v} \in \mathbb{R}^m$, $\boldsymbol{u} \in \mathbb{R}^{nd}$, $\boldsymbol{\lambda} \in \mathbb{R}^c$ and $\boldsymbol{r} \in \mathbb{R}^{nd}$. Moreover, M will be assumed to be symmetric, positive definite, and the rows of C will generally be assumed to be linearly independent (that is, unless otherwise mentioned there are no redundant equality constraints).

2.3.1 Reduced formulation

We can eliminate the variable $\boldsymbol{v}$ from the DCFP (2.17), and obtain

$$\begin{cases} P\boldsymbol{\lambda} + B^\intercal\boldsymbol{r} + \boldsymbol{c} = \boldsymbol{0} \\ B\boldsymbol{\lambda} + W\boldsymbol{r} + \boldsymbol{b} = \boldsymbol{u} \\ (\boldsymbol{u}_i, \boldsymbol{r}_i) \in \mathscr{C}_{\mu_i} \qquad \forall 1 \le i \le n. \end{cases} \tag{2.18}$$

with

$$W := HM^{-1}H^T \qquad P := CM^{-1}C^T \qquad B := HM^{-1}C^T$$
$$\boldsymbol{b} := HM^{-1}\boldsymbol{f} + \boldsymbol{w} \qquad \boldsymbol{c} := CM^{-1}\boldsymbol{f} - \boldsymbol{k}.$$

The symmetric, positive semi-definite matrix W is often referred to as the *Delassus* operator.

When the rows of C are linearly independent, C is surjective, therefore P is invertible and we can go one step further, writing system (2.18) with $\boldsymbol{r}$ and $\boldsymbol{u}$ as the only unknowns,

$$\begin{cases} \boldsymbol{u} = \tilde{W}\boldsymbol{r} + \tilde{b} \\ (\boldsymbol{u}_i, \boldsymbol{r}_i) \in \mathscr{C}_{\mu_i} \qquad \forall 1 \le i \le n. \end{cases} \tag{2.19}$$

with $\tilde{W} := W - BP^{-1}B^\intercal$ and $\tilde{b} := \boldsymbol{b} - BP^{-1}\boldsymbol{c}$.

While this formulation is short and convenient, in practice we will often avoid computing the $\tilde{W}$ matrix explicitly. First, the inverse of M may be dense; this will be the case for cloth, or finite-element models, among others. In this case, W and P will be dense, and therefore expensive to

compute and less likely to fit into main memory. It is also rarely a good idea to compute $\tilde{W}$, as P^{-1} will often be dense and the double-inversion might lead to tremendous numerical errors.

However, for multi-body systems, M is a block-diagonal matrix; if the size of the diagonal blocks is reasonable, they can be easily factorized, for instance using a $QLDL^\intercal Q^\intercal$ factorization [4], and it may become fruitful to explicitly compute the Delassus operator. Even then, W might be costly to assemble. We argue in (Daviet, Bertails-Descoubes, and Boissieux 2011) that it is important to exploit the block structure of M and H to accelerate the computation. Our `bogus` (Daviet 2013) C++ template library provides an easy way to perform efficient linear algebra operations with sparse block matrices, including a parallelized matrix–matrix multiplication routine. Despite this, our results for hair simulations (presented in the next chapter, Table 4.2) show that a significant portion of the total time for solving the DCFP (2.17) was spent assembling the Delassus matrix W, and (Daviet, Bertails-Descoubes, and Boissieux 2011, Figure 6) implies a scaling with the square of the number of contacts, both in time and memory consumption.

More generally, a trade-off has to be made between a higher preprocessing and memory cost, or a higher solver cost. Some of the algorithms that we will present in Chapter 3 will require an explicit computation of W, others will just require being able to perform matrix–vector multiplications with W, some will use completely different formulations; their relative performance is application-dependent.

2.3.2 Fixed-point algorithms and existence criterion

Still following from the fact that the Coulomb friction law does not derive from a potential, the DCFP (2.17) cannot be formulated as a convex optimization problem. However, (2.17) can be characterized as the fixed-point of sequences of such problems; we just need to temporarily fix either $\boldsymbol{u}_i$ or $\boldsymbol{r}_i$ in the coupling term of the bipotential (1.14), $\mu_i r_{\mathrm{iN}} \|\boldsymbol{u}_{\mathrm{iT}}\|$.

As the minimization of convex functions is a well-studied problem, for which numerous efficient algorithms have been devised, this iterative strategy may prove fruitful in practice. Moreover, this characterization of (2.17) as a fixed-point was used by Cadoux (2009) to state a sufficient criterion for the existence existence of solutions, which we will recall in Theorem 2.1.

Haslinger algorithm First, let us fix the normal force $\boldsymbol{r}_{\mathrm{iN}}$. Consider the function h_s (which is no-longer a bipotential) derived from (1.14) as

$$h_s(\boldsymbol{u}_i, -\boldsymbol{r}_i) = \mathscr{I}_{-\mathscr{K}_{\mu_i}}(-\boldsymbol{r}_i) + \mathscr{I}_{\mathscr{K}_\infty}(\boldsymbol{u}_i) + s_i \|\boldsymbol{u}_{\mathrm{iT}}\|.$$

A velocity–force couple satisfying $\boldsymbol{r}_i \in -\partial h_s(\boldsymbol{u}_i)$ will be a solution to the Coulomb friction problem if and only if $s_i = \mu_i r_{\mathrm{iN}}$. Now, we recognize from Equation (1.25) the expression of the convex conjugate of $\mathscr{I}_{\mathscr{T}_{0,s_i}}$, that this, $\mathscr{I}_{\mathscr{K}_\infty}(\boldsymbol{u}_i) + s_i\|\boldsymbol{u}_{\mathrm{iT}}\| = \mathscr{I}_{\mathscr{T}_{0,s_i}}{}^\star(\boldsymbol{u}_i)$ (note that $\mathscr{T}_{0,s_i}$ is simply a semi-infinite cylinder of radius s_i). This means, using Theorem A.2,

$$\boldsymbol{r}_i \in -\partial h_s(\boldsymbol{u}_i) \iff \boldsymbol{r}_i \in -\partial\left(\mathscr{I}_{\mathscr{T}_{0,s_i}}{}^\star\right)(\boldsymbol{u}_i) \iff \boldsymbol{u}_i \in -\mathscr{N}_{\mathscr{T}_{0,s}}(\boldsymbol{r}_i).$$

As the optimality conditions of the quadratic minimization problem under cylindrical constraints (2.20),

$$\min_{\boldsymbol{\lambda}\in\mathbb{R}^c, \boldsymbol{r}\in\Pi\mathscr{T}_{0,s_i}} \frac{1}{2}(\boldsymbol{\lambda}^\intercal, \boldsymbol{r}^\intercal)\begin{pmatrix} P & B^\intercal \\ B & W \end{pmatrix}\begin{pmatrix} \boldsymbol{\lambda} \\ \boldsymbol{r} \end{pmatrix} + (\boldsymbol{\lambda}^\intercal, \boldsymbol{r}^\intercal)\begin{pmatrix} \boldsymbol{c} \\ \boldsymbol{b} \end{pmatrix}, \tag{2.20}$$

are, from Theorem (A.6),

$$\begin{cases} P\boldsymbol{\lambda} + B^\intercal \boldsymbol{r} + \boldsymbol{c} = \boldsymbol{0} \\ B\boldsymbol{\lambda} + W\boldsymbol{r} + \boldsymbol{b} = \boldsymbol{u} \\ \boldsymbol{u}_i \in -\mathscr{N}_{\mathscr{T}_{0,s_i}}(\boldsymbol{r}_i) \quad \forall i, \end{cases}$$

[4] That is, a Cholesky factorization where the square-root of the diagonal is not explicitly computed for better numerical precision, and with a reordering step to reduce the fill-in of the triangular matrix L. The `SimplicialLDLT` class of the Eigen (Guennebaud, Jacob, et al. 2010) C++ library provides an implementation of this factorization.

we deduce that the solution of the minimization problem (2.20) will be the solution of the dual form of the DCFP (2.18) if and only if $\boldsymbol{s} := (s_i) = (\mu_i \boldsymbol{r}_{\mathrm{iN}})$. This leads to the fixed-point algorithm introduced by Haslinger (1983), which consists in iteratively solving the minimization problem (2.20)[5] for a given parameter value $\boldsymbol{s}^k$, then computing $\boldsymbol{s}^{k+1}$ from the optimal solution, and looping again.

However, the objective function of (2.20) is not strictly convex, and therefore the problem may have more than one solution. The fixed-point iteration is thus ill-defined, and studying its convergence would be tricky. In contrast, Cadoux (2009) formulated an algorithm that does not suffer from this drawback, by freezing instead the velocity part of the bipotential coupling term.

Cadoux algorithm Let us now consider the function derived from the bipotential by fixing $\boldsymbol{u}$ in the coupling term,

$$c_s(\boldsymbol{u}_i, -\boldsymbol{r}_i) := \mathscr{I}_{-\mathscr{K}_{\mu_i}}(-\boldsymbol{r}_i) + \mathscr{I}_{\mathscr{K}_\infty}(\boldsymbol{u}_i) + s_i \boldsymbol{r}_{\mathrm{iN}}.$$

The equivalence $(\boldsymbol{u}_i, \boldsymbol{r}_i) \in \mathscr{C}_{\mu_i} \iff \boldsymbol{u}_i \in \frac{\partial c_s}{\partial -\boldsymbol{r}_i}(-\boldsymbol{r}_i)$ will hold if and only if $s_i = \mu_i \|\boldsymbol{u}_{\mathrm{iT}}\|$. Now, direct computations yield that $\boldsymbol{u}_i \in \frac{\partial c_s}{\partial -\boldsymbol{r}_i}(-\boldsymbol{r}_i) \iff \boldsymbol{u}_i + s_i \boldsymbol{n}_i \in -\mathscr{N}_{\mathscr{K}_{\mu_i}}(\boldsymbol{r}_i)$. The solution of the DCFP (2.18) will therefore be given by the solution of the minimization problem (2.21),

$$q(\boldsymbol{s}) := \min_{\boldsymbol{\lambda} \in \mathbb{R}^c, \boldsymbol{r} \in \Pi \mathscr{K}_{\mu_i}} \frac{1}{2} (\boldsymbol{\lambda}^\intercal, \boldsymbol{r}^\intercal) \begin{pmatrix} P & B^\intercal \\ B & W \end{pmatrix} \begin{pmatrix} \boldsymbol{\lambda} \\ \boldsymbol{r} \end{pmatrix} + (\boldsymbol{\lambda}^\intercal, \boldsymbol{r}^\intercal) \begin{pmatrix} \boldsymbol{c} \\ \boldsymbol{b} + \boldsymbol{s} \end{pmatrix}, \tag{2.21}$$

when $\boldsymbol{s} = (\mu_i \|\boldsymbol{u}_{\mathrm{iT}}\| \boldsymbol{n}_i)$, with $\boldsymbol{u} := W\boldsymbol{r} + B\boldsymbol{\lambda} + \boldsymbol{b}$

We will refer to problems such as (2.21), where a quadratic objective function is minimized subject to SOC constraints, as Second-Order Cone Quadratic Programs (SOCQP). A fixed-point algorithm can then be obtained by computing $\boldsymbol{s}^{k+1}$ from the solution λ^k to the problem parameterized by $\boldsymbol{s}^k$ as $\boldsymbol{s}^{k+1} = (\mu_i \|\boldsymbol{u}^k_{\mathrm{iT}}\|)$.

At first sight, the gain over the Haslinger algorithm is not obvious; the solution to the minimization problem (2.21) is still not unique. Cadoux (2009, Theorem 3.5) proves the well-definition of the fixed-point update from a duality argument, using Fenchel's duality theorem (A.3) to show that the velocity $\boldsymbol{v}$ corresponding to the optimum of (2.21) is the solution of a strictly convex minimization problem, meaning that $\boldsymbol{v}$ and thus $\boldsymbol{u}$ are uniquely defined for each $\boldsymbol{s}$. Here, we will show this well-definition by directly constructing the primal problem from the optimality condition of the dual (2.21).

Expressing W and P in the optimality conditions of (2.21), we can get back to a system in the same unknowns as our original DCFP (2.17),

$$\begin{cases} M\boldsymbol{v} = \boldsymbol{f} + C^\intercal \boldsymbol{\lambda} + H^\intercal \boldsymbol{r} \\ C\boldsymbol{v} = \boldsymbol{k} \\ \quad \boldsymbol{u} = H\boldsymbol{v} + \boldsymbol{w} \\ \mathscr{K}_{\frac{1}{\mu_i}} \ni \boldsymbol{u}_i + \boldsymbol{s}_i \boldsymbol{n}_i \perp \boldsymbol{r}_i \in \mathscr{K}_{\mu_i} \quad \forall i = 1 \dots n. \end{cases} \tag{2.22}$$

Since system (2.22) derives from the optimality conditions of a convex problem, we we will refer to (2.22) as a *convexified* DCFP. Now, discarding the relative velocity variable $\boldsymbol{u}$, system (2.22) can be written equivalently as

$$\begin{cases} M\boldsymbol{v} = \boldsymbol{f} + C^\intercal \boldsymbol{\lambda} + H^\intercal \boldsymbol{r} \\ \quad \boldsymbol{\lambda} \in -\mathscr{N}_{\{\boldsymbol{0}_{\mathbb{R}^c}\}}(C\boldsymbol{v} - \boldsymbol{k}) \\ \quad \boldsymbol{r} \in -\mathscr{N}_{\Pi \mathscr{K}_{\frac{1}{\mu_i}}}(H\boldsymbol{v} + \boldsymbol{w} + \boldsymbol{s}). \end{cases}$$

[5] Actually, the original Haslinger algorithm solves the Fenchel primal of (2.20).

Moreover, from Corollary A.4 to the Property A.12 on the subdifferential of a precomposition with an affine map, we have always

$$C^\intercal \mathcal{N}_{\{\mathbf{0}_{\mathbb{R}^c}\}}(C\boldsymbol{v}-\boldsymbol{k}) + H^\intercal \mathcal{N}_{\Pi\mathscr{K}_{\frac{1}{\mu_i}}}(H\boldsymbol{v}+\boldsymbol{w}+\boldsymbol{s}) \subset \mathcal{N}_{V(\boldsymbol{s})}(\boldsymbol{v})$$

with $V(\boldsymbol{s}) := \left\{\boldsymbol{v}\in\mathbb{R}^m, C\boldsymbol{v}=\boldsymbol{k} \text{ and } (H\boldsymbol{v}+\boldsymbol{w}+\boldsymbol{s})\in\Pi\mathscr{K}_{\frac{1}{\mu_i}}\right\}$, and equality is achieved under the regularity condition $(\mathscr{H}(\boldsymbol{s}))$,

$$\exists \boldsymbol{v}\in\mathbb{R}^m, C\boldsymbol{v}=\boldsymbol{k} \text{ and } (H\boldsymbol{v}+\boldsymbol{w}+\boldsymbol{s})\in \operatorname{int}\Pi\mathscr{K}_{\frac{1}{\mu_i}}. \qquad (\mathscr{H}(\boldsymbol{s}))$$

This means that all the solutions to (2.22) must satisfy (2.23),

$$M\boldsymbol{v}-\boldsymbol{f} \in -\mathcal{N}_{V(\boldsymbol{s})}(\boldsymbol{v}) \qquad (2.23)$$

and that they coincide when the condition $(\mathscr{H}(\boldsymbol{s}))$ is satisfied. Finally, we recognize (2.23) as the optimality conditions of the strictly convex minimization problem (2.24),

$$p(\boldsymbol{s}) := \min_{\boldsymbol{v}\in V(\boldsymbol{s})} \frac{1}{2}\boldsymbol{v}^\intercal M\boldsymbol{v} - \boldsymbol{v}^\intercal \boldsymbol{f}. \qquad (2.24)$$

Summarizing, the velocity $\boldsymbol{v}$ reconstructed from a solution $(\boldsymbol{\lambda},\boldsymbol{r})$ of the dual SOCQP (2.21) as $\boldsymbol{v}=M^{-1}(\boldsymbol{f}+C^\intercal\boldsymbol{\lambda}+H^\intercal\boldsymbol{r})$ will always be a solution of the primal SOCQP (2.24), and if $\boldsymbol{v}$ is a solution to the primal (2.24), then the condition $(\mathscr{H}(\boldsymbol{s}))$ suffices to ensure the existence of a solution $(\boldsymbol{\lambda},\boldsymbol{r})$ to (2.21).

Property 2.1 (Cadoux fixed-point algorithm)**.** *We introduce the mapping $s:\mathbb{R}^{nd}\to\mathbb{R}^{nd}$, $(\boldsymbol{u}_i)\mapsto(\mu_i\|\boldsymbol{u}_{iT}\|\boldsymbol{n}_i)$. Let $v(\boldsymbol{s})$ denote the optimum of the strictly-convex primal SOCQP (2.24), and $u:\boldsymbol{v}\mapsto\boldsymbol{u}:=H\boldsymbol{v}+\boldsymbol{w}$. Then, if $V(\mathbf{0})\neq\emptyset$, the mapping $F:\mathbb{R}^{nd}\to\mathbb{R}^{nd}$, $F:=s\circ u\circ v$ is well-defined on $\operatorname{Im}s\supset\operatorname{Im}F$, and under the condition $\mathscr{H}(\mathbf{0})$, any fixed-point of F will correspond to a solution of the DCFP (2.17).*

The sequence $\boldsymbol{s}^0 := s\circ u(\boldsymbol{v}^0)$, $\boldsymbol{s}^{k+1}:=F(\boldsymbol{s}^k)$ will be referred to as the Cadoux fixed point algorithm.

Proof. Notice that $\forall i\in 1\dots n$, $s(\boldsymbol{u})_i\in\mathscr{K}_0$; this means that for any $\boldsymbol{s}$ in the image of s, $\forall\boldsymbol{z}\in\Pi\mathscr{K}_{\frac{1}{\mu_i}}$, $(\boldsymbol{z}+\boldsymbol{s})\in\mathscr{K}_{\frac{1}{\mu_i}}$ and $\forall\boldsymbol{z}\in\operatorname{int}\Pi\mathscr{K}_{\frac{1}{\mu_i}}$, $(\boldsymbol{z}+\boldsymbol{s})\in\operatorname{int}\mathscr{K}_{\frac{1}{\mu_i}}$. Therefore, for all $\boldsymbol{s}\in\operatorname{Im}s$, $V(\boldsymbol{s})\subset V(\mathbf{0})$, and $\mathscr{H}(\mathbf{0})\implies\mathscr{H}(\boldsymbol{s})$.

If $V(\mathbf{0})\neq\emptyset$, the primal SOCQP (2.24) is thus feasible for any $\boldsymbol{s}\in\operatorname{Im}s$, and as it is strictly-convex it admits a unique solution. v is therefore well-defined on $\operatorname{Im}s$, and so is F.

Now, let $\bar{\boldsymbol{s}}$ be a fixed-point of F. Under the condition $\mathscr{H}(\mathbf{0})$, $\mathscr{H}(\bar{\boldsymbol{s}})$ is satisfied and therefore $v(\bar{\boldsymbol{s}})$ will correspond to a solution $(\bar{\boldsymbol{\lambda}},\bar{\boldsymbol{r}})$ of the dual SOCQP (2.21) at $\bar{\boldsymbol{s}}$. As $\bar{\boldsymbol{s}}=F(\bar{\boldsymbol{s}})=(\mu_i\|\boldsymbol{u}_{iT}\|\boldsymbol{n}_i)$, $(\bar{\boldsymbol{\lambda}},\bar{\boldsymbol{r}})$ is also a solution of the dual for of the DCFP (2.17). □

Cadoux (2009) also investigated other update rules for $\boldsymbol{s}$, for instance using a Newton or quasi-Newton iteration instead of a fixed-point one. Those more complex rules were not found to perform significantly better in practice, and we will not consider them here.

In practice $v(\boldsymbol{s})$ may be computed either from the solution to the primal SOCQP (2.24), or from the dual (2.21). The primal has the advantage of boasting strict convexity, but is subject to a more complex constraint, as it involves the linear application H on top of the SOC. For some values of $\boldsymbol{s}$, $p(\boldsymbol{s})$ might admit a solution while $q(\boldsymbol{s})$ does not; however, an approximate solution for $q(\boldsymbol{s})$ might suffice to keep the algorithm going. Actually, it is often a good idea to truncate the resolution of the intermediate SOCQP, using heuristics to refine error tolerance as the fixed-point algorithm converges, to increase the computational performance of the algorithm as a whole.

Cadoux (2009) proves furthermore — in the case without equality constraints, but the generalization is easy — the following theorem about the existence of solutions:

Theorem 2.1 (Cadoux existence criterion)**.** *If $V(\mathbf{0}) \neq \emptyset$, then the mapping F introduced in Property 2.1 admits a fixed-point $\bar{\boldsymbol{s}}$. If $\mathscr{H}(\bar{\boldsymbol{s}})$, or a fortiori if $\mathscr{H}(\mathbf{0})$ is satisfied, then the DCFP (2.17) admits a solution.*

Note that Theorem 2.1 gives a *sufficient* existence condition, but not a *necessary* one. For instance, Cadoux (2009, Theorem 3.20) states that a solution to DCFP (2.17) will exist as soon as the fixed-point $\bar{\boldsymbol{s}}$ is such that the constraints for which $\bar{\boldsymbol{s}}_i = \mathbf{0}$ satisfy a qualification condition.

Anistescu regularization Using the regularized contact law proposed by Anitescu (2005) amounts to approximating the DCFP (2.17) with the convexified DCFP

$$\begin{cases} M\boldsymbol{v} = \boldsymbol{f} + C^\mathsf{T}\boldsymbol{\lambda} + H^\mathsf{T}\boldsymbol{r} \\ C\boldsymbol{v} = \boldsymbol{k} \\ \boldsymbol{u} = H\boldsymbol{v} + \boldsymbol{w} \\ \mathscr{K}_{\frac{1}{\mu_i}} \ni \boldsymbol{u}_i \perp \boldsymbol{r}_i \in \mathscr{K}_{\mu_i} \qquad \forall i = 1 \dots n, \end{cases}$$

which corresponds to the optimality conditions of our couple of dual SOCQP at $\boldsymbol{s} = \mathbf{0}$. Using the Drucker–Prager analogy (Section 1.3.3), this corresponds to adding a dilatancy with coefficient $\frac{1}{\mu}$ to the Coulomb friction law, i.e., in plasticity terms, using an associated flow rule. Mazhar et al. (2015) argue that numerical simulations using this regularization are still able to match experimental data on several complex examples. However we can readily see that the approximation becomes rougher for higher values of the friction coefficient or of the tangential relative velocity, and cause the objects to separate instead of sliding.

Summary

In this chapter, we have seen that the equations of motion of a large class of mechanical systems subject to holonomic kinematic constraints and unilateral contact with Coulomb friction can be integrated in time by solving a sequence of problems structurally similar to the Discrete Coulomb Friction Problem (2.17). The DCFP itself can then be solved as a sequence of Second-Order Cone Quadratic Programs, using either the primal (2.24) or dual (SOCQP (2.21) formulation.

In the following chapter, we will present numerical algorithms for solving either:

- the original DCFP (2.17);
- the convexified DCFP (2.22), or the corresponding primal (2.24) and dual (2.21) SOCQP.

3 Solving the Discrete Coulomb Friction Problem

This chapter will present an overview of different classes of methods that have been proposed in the literature to solve Discrete Coulomb Friction Problems, in both their standard and convexified versions. We briefly recall the differences between these two problems:

- In the case of the standard DCFP (2.17), each pair of local relative velocity and contact force $(\boldsymbol{u}_i, \boldsymbol{r}_i)$ must satisfy the Coulomb law, i.e., $(\boldsymbol{u}_i, \boldsymbol{r}_i) \in \mathscr{C}_{\mu_i}$. This is equivalent to $\mathscr{K}_{\frac{1}{\mu_i}} \ni \tilde{\boldsymbol{u}}_i \perp \boldsymbol{r}_i \in \mathscr{K}_{\mu_i}$, where $\tilde{\boldsymbol{u}}_i$ is defined by the *de Saxcé* change of variable from Property 1.4, $\tilde{\boldsymbol{u}}_i := \boldsymbol{u}_i + \mu_i \|\boldsymbol{u}_{i\mathrm{T}}\| \boldsymbol{n}$.
- In the convexified DCFP (2.17), each pair of local relative velocity and contact force $(\boldsymbol{u}_i, \boldsymbol{r}_i)$ must directly satisfy the complementarity relationship $\mathscr{K}_{\frac{1}{\mu_i}} \ni \boldsymbol{u}_i \perp \boldsymbol{r}_i \in \mathscr{K}_{\mu_i}$. The convexified DCFP is equivalent to a minimization problem, the dual SOCQP (2.21), and, under the regularity condition $\mathscr{H}(\mathbf{0})$, to the primal SOCQP (2.24).

We recall also that the standard DCFP can be recast as a sequence of convexified DCFP using the Cadoux (2009) fixed-point algorithm from Section 2.3.2.

There has been a considerable amount of research dedicated to solving DCFP in the last decades, and to the best of our knowledge no comprehensive review exists. Such is not the goal of this chapter either; what follows is an opinionated presentation of the methods that we found to be of practical or historical interest, and we refer the reader to (Acary and Brogliato 2008; Cadoux 2009; Heyn 2013) for a larger covering of the literature. In the next chapter, we will also present an algorithm designed to be robust to ill-conditioned problems, and that performed well on a few applications.

To lighten notations and when no confusion is possible, we will use $\mathscr{K}_\mu$ ($\mathscr{K}_{\frac{1}{\mu}}$) to designate indifferently the SOC in $\mathbb{R}^d$ and $\Pi_i \mathscr{K}_{\mu_i}$ ($\Pi_i \mathscr{K}_{\frac{1}{\mu_i}}$), the product of n SOC in $\mathbb{R}^{nd}$, .

3.1 Global strategies

3.1.1 Pyramidal friction cone

In 2D, the SOC can be described by two linear constraints. This is no longer the case in 3D, however approximating the 3D SOC with a pyramid (using a *facetting* process) is a popular way of writing the friction problem with only a set of linear constraints.

One can define an approximation of the friction cone as q-sided pyramid $\mathscr{P}_\mu^{p,\boldsymbol{d}}$,

$$\mathscr{P}_\mu^{p,\boldsymbol{d}} := \left\{ \boldsymbol{r} \in \mathbb{R}^d,\ \left\| \left(\langle \boldsymbol{r}, \boldsymbol{e}_j \rangle \right)_{1 \le j \le q} \right\|_p \le \mu r_{\mathrm{N}} \right\},\ \boldsymbol{e}_j := \frac{\boldsymbol{d}_j}{\left\| \left(\langle \boldsymbol{d}_j, \boldsymbol{d}_k \rangle \right)_{1 \le k \le q} \right\|_p}$$

where $(\boldsymbol{d}_j)$ is a set of q unit vectors in the tangent plane, and $p = 1$ (inner approximation) or $+\infty$ (outer approximation). In practice, it is is common to choose $(\boldsymbol{d}_j)$ as $d-1$ orthogonal vectors. In order to minimize the effect of the linearization of the cone, it is advocated to choose the vectors in $(\boldsymbol{d}_j)$ such that the faceted cone coincides with the exact one in the direction of the relative velocity at the beginning of the timestep.

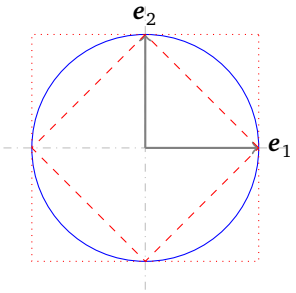


Figure 3.1: *Horizontal section of the inner (red, dashed) and outer (red, dotted) approximations with two basis vectors of the 3D Coulomb cone (blue).*

The faceted problem can then be solved using a variety of methods; we will not detail them, as our main goal will be to solve the problem in 3D (or more) with the exact cone. We refer to (Erleben 2013) for a recent review such algorithms. In a non-exhaustive manner, we can mention:

- the Lemke algorithm used by Klarbring (1987), a direct method with good convergence properties, but which may become expensive for large systems;
- a conjugate-gradient algorithm on a sequence of minimization problems (Renouf and Alart 2005);
- the staggered projection approach (Kaufman, Sueda, et al. 2008), popular in Computer Graphics, which solves for the normal and tangential components in an iterative manner;
- an interior-point method (Trinkle et al. 1997);
- the Gauss-Seidel splitting algorithm (e.g., Raous et al. 1988, Jean 1999, Erleben 2007), which we will detail in Section (3.4).

This faceting approach also possesses weaknesses; for instance, Renouf, Acary, et al. (2005) argue that this approximation may lead to significant errors in the resulting trajectories.

Moreover, while the structure of the individual constraints is indeed simpler, their number is drastically increased (the simplest approximation, a 4-sided pyramid, already requires 4 linear constraints and 2 dual variables, instead of a single quadratic constraint and a single dual variable for the exact cone). The constraints also become more prone to switching between active and inactive states (the number of faces is increased), which might degrade the overall convergence of the solver algorithm. Overall, the faceted problem might therefore not be simpler nor faster to solve than the original one[1], and the pertinence of this approximation should be carefully evaluated.

For these reasons, in the remainder of this chapter we will focus on the original, non-linearized problem defined with forces inside a SOC.

3.1.2 Complementarity functions

A natural way of solving the reduced dual formulation (2.19) is to express Coulomb law as a the root of a complementarity function, for instance the Alart–Curnier function (1.6) or the De Saxcé (1.29) complementarity functions. In the next chapter, we will also present another complementarity function, based on a adaptation of the Fischer–Burmeister function to the SOC algebra (Fukushima et al. 2002). Note that while the De Saxcé complementarity function can easily be adapted to solve the convexified DCFP rather than the DCFP (by simply not performing the eponymous change of variable), it is not as simple in the case of the Alart–Curnier function.

[1] Rockafellar (1993) famously said that "the great watershed in optimization isn't between linearity and nonlinearity, but convexity and nonconvexity".

In a general manner, let $f : \mathbb{R}^{nd} \times \mathbb{R}^{nd} \to \mathbb{R}^{nd}$ be such that $f(\boldsymbol{u}, \boldsymbol{r}) = \boldsymbol{r} \iff (\boldsymbol{u}_i, \boldsymbol{r}_i) \in \mathscr{C}_{\mu_i} \forall i$. Then finding a solution to (2.19) amounts to finding $\boldsymbol{r} \in \mathbb{R}^{nd}$ such that

$$f(\tilde{W}\boldsymbol{r} + \tilde{b}, \boldsymbol{r}) = \boldsymbol{r}. \tag{3.1}$$

As it now involves solely the force variable $\boldsymbol{r}$, a variety of algorithms may be used to to find a solution to (3.1); however, remember that f is in general non-convex, non-contracting, and non-smooth.

Using a simple fixed-point iteration $\boldsymbol{r}^{k+1} = f(\tilde{W}\boldsymbol{r}^k + \tilde{b}, \boldsymbol{r}^k)$ is tempting, and has been used in the case of a single contact (e.g., Feng et al. 2005) [2] , but the convergence is not always satisfying for larger systems in practice. As the points at which f is non-differentiable are generally sparse, a better solution is to use a Newton algorithm[3] on the function g (Alart and Curnier 1991; Dumont 2012), $\boldsymbol{g} : \mathbb{R}^{nd} \to \mathbb{R}^{nd}$, $\boldsymbol{r} \mapsto f(\tilde{W}\boldsymbol{r} + \tilde{b}, \boldsymbol{r}) - \boldsymbol{r}$. Each step is defined as the solution to a linear system,

$$J_g^k \left(\boldsymbol{r}^{k+1} - \boldsymbol{r}^k\right) = -\alpha^k \boldsymbol{g}(\boldsymbol{r}^k)$$

where $J_g^k \in \partial g(\boldsymbol{r}^k)$ is any element of the subdifferential of $\boldsymbol{g}$ at $\boldsymbol{r}^k$, and α^k is an optional damping term that can be computed using a line-search based on the error function $\|g\|^2$.

Drawbacks The left-hand-side matrix of the linear system that must be solved at each step,

$$\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{r}}(\boldsymbol{r}^k) = \tilde{W}\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}}(\tilde{W}\boldsymbol{r}^k + \tilde{b}) + \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{r}}(\boldsymbol{r}^k) - \mathbb{I}_{\mathbb{R}^{nd}},$$

may not be invertible. A failsafe is required when one cannot find a step increment, for instance a gradient-descent or fixed-point iteration. In any case, the cost of solving this linear system grows quickly with the number of contacts, making the cost of this algorithm prohibitive for very large systems.

Moreover, we have observed in (Bertails-Descoubes et al. 2011) that while a Newton algorithm on the Alart-Curnier formulation yields good results for a system with a small number of contacts, the success rate of the algorithm drops quickly when a conditioning parameter, $\mu\frac{nd}{m}$, grows beyond 1. Indeed, in such cases the number of rows of H exceeds its number of columns, and the number of vanishing eigenvalues of $\tilde{W}$ increases. Moreover, criterion $\mathscr{H}(\boldsymbol{s})$ implies a link between the surjectivity of H and the existence of a solution to the DCFP; as the relative rank of H degrades, so does the probability that this existence criterion is satisfied.

3.1.3 Optimization-based methods

Since the convexified DCFP can be recast as either the primal (2.24) or dual (2.21) SOCQP, it is natural to attempt to use optimization-based algorithms to solve this problem. Moreover, as a SOCQP consists in the minimization of a convex function over a convex feasible set, theoretical convergence properties can generally be derived.

In the following sections, we will present three kinds of optimization-based methods: interior-point methods, proximal algorithms, and splitting algorithms. The latter family of methods can be interpreted as block-coordinate-descent algorithms, i.e., partial minimization over a small number of variables in turn.

Optimization-based methods can be leveraged to solve the original DCFP thanks to the Cadoux fixed-point algorithm defined in Property 2.1. Alternatively, we will see in the following that some optimization-based algorithms (e.g., projected gradient descent or Gauss–Seidel) can be repurposed to directly solve the original DCFP. However, both strategies induce the loss of any theoretical convergence property.

[2] When f is a projection function, i.e., $f \sim \Pi_C(\boldsymbol{r} - \rho\boldsymbol{u})$, as is the case for the Alart-Curnier and De Saxcé complementarity function, the fixed-point algorithm can be interpreted as a slight variation of a proximal algorithm; see Feng et al. 2005 and Section 3.3 below.

[3] Quasi-Newton methods on complementarity functions are rarely observed to perform well in practice.

3.2 Interior-point methods

Interior-point methods have become extremely popular in optimization communities in the last decades, thanks to their superlinear convergence on many problems (Potra and Wright 2000).

At their core, they replace the characteristic function $\mathscr{I}_C$ associated to the constraint C with a logarithmic *barrier* function, $\mathscr{I}_{C,\beta}$, such that $\lim_{\beta\to 0}\mathscr{I}_{C,\beta} = \mathscr{I}_C$. If the feasible set C is defined as $C = \{\boldsymbol{x}, F(\boldsymbol{x}) \leq 0\}$, one may take $\mathscr{I}_{C,\beta}(\boldsymbol{x}) := -\beta\log(-F(\boldsymbol{x}))$. The interior-point method then performs a Newton algorithm on the modified optimization problem, while simultaneously driving β to 0 in the process.

Note that high values for β will tend to make the successive iterates stay in the "center" of the feasible set. Iterates will thus follow an interior path to reach the solution, even when this optimal point lies on the boundary of the constraint, which is where the interior-point method's name come from. This is in contrast to most other kinds of methods (pivoting, active-set, proximal, ...) which tend instead to follow the boundary of the constraint.

3.2.1 Second-Order Cone Programs

A particular case for which several efficient solvers[4] have been devised is Second Order Cone Programs (SOCP), that is, minimizing a linear objective function under equality and SOC constraints. Our convexified DCFP corresponds to the optimality conditions of a (primal or dual) SOCQP, and this class of problem can easily be recast as SOCP, as shown in Cadoux (2009) (more generally, quadratic optimization problems with quadratic constraints are a subclass of SOCP, see e.g., E. Andersen and K. Andersen 2013). The trick is in the transformation of the quadratic objective function into a linear objective function with a SOC constraint. Indeed, the problem

$$\min_{\boldsymbol{x}\in C}\frac{1}{2}\boldsymbol{x}^\intercal M\boldsymbol{x} + \boldsymbol{x}^\intercal\boldsymbol{b}$$

is equivalent to

$$\begin{cases}\min\limits_{\boldsymbol{x}\in C, t\in\mathbb{R}} & t + \boldsymbol{x}^\intercal\boldsymbol{b}\\ & 2t \geq \boldsymbol{x}^T M\boldsymbol{x}.\end{cases} \tag{3.2}$$

Given a square root L of M (for instance, its Cholesky factorization), Problem (3.2) becomes

$$\begin{cases}\min\limits_{\boldsymbol{x}\in C, t\in\mathbb{R}} & t + \boldsymbol{x}^\intercal\boldsymbol{b}\\ & L^\intercal\boldsymbol{x} = \boldsymbol{z}\\ & 2t \geq \|\boldsymbol{z}\|^2.\end{cases}$$

The constraint $t \geq \|\boldsymbol{z}\|^2$ can then be recognized as the rotated cone constraint $(1, t, \boldsymbol{z}) \in \mathscr{RK}$, i.e., $2\times 1\times t \geq \|\boldsymbol{z}\|^2$. Rotated cone constraints are handled by most SOCP solvers, or can be recast as standard SOC constraints.

The SOCP corresponding to the primal SOCQP (2.24) is therefore

$$\begin{cases}\min\limits_{\boldsymbol{v}\in\mathbb{R}^m, t\in\mathbb{R}} & t - \boldsymbol{v}^\intercal\boldsymbol{f}\\ & \boldsymbol{z} = L^\intercal\boldsymbol{v}\\ & \boldsymbol{k} = C\boldsymbol{v}\\ & \boldsymbol{u} = H\boldsymbol{v} + \boldsymbol{x} + \boldsymbol{s}\\ & \boldsymbol{u}_i \in \mathscr{K}_{\frac{1}{\mu_i}} \qquad \forall i = 1\dots n\\ & (1, t, \boldsymbol{z}) \in \mathscr{RK},\end{cases} \tag{3.3}$$

[4] Including several out-of-the-box commercial packages, such as CPLEX, MOSEK, Gurobi, ...

and the dual SOCQP (2.21) can be in turn transformed into

$$\begin{cases} \min\limits_{\boldsymbol{r}\in\mathbb{R}^{nd},\boldsymbol{\lambda}\in\mathbb{R}^{c},t\in\mathbb{R}} & t+\boldsymbol{r}^{\mathsf{T}}(\boldsymbol{b}+\boldsymbol{s})+\boldsymbol{\lambda}^{\mathsf{T}}\boldsymbol{c} \\ & L^{\mathsf{T}}\boldsymbol{z}=H^{\mathsf{T}}\boldsymbol{r}+C^{\mathsf{T}}\boldsymbol{\lambda} \\ & \boldsymbol{r}_i\in\mathscr{K}_{\mu_i} \qquad\qquad \forall i=1\dots n \\ & (1,t,\boldsymbol{z})\in\mathscr{R}\mathscr{K} \end{cases} \tag{3.4}$$

with the square-root matrix L such that $M=LL^{\mathsf{T}}$.

Note that this technique can only be applied to the convexified DCFP; there is no hope of casting the original DCFP into a SOCP.

3.2.2 Discussion

Warm-starting A major drawback of interior-point methods is that they are hard to warm-start; that is, the solution to a similar (or even to the same) optimization problem will not be able to serve as an initial guess for a subsequent solve.

Indeed, while the solution from a previous solve may be arbitrarily close to the boundary of the constraint, most interior-point algorithms start with a relatively high value for the barrier parameter β. This will make the previous solution very poor (that is, yielding a very high value of the objective function), and the next iterate will go back towards the center of the constraint.

This drawback is especially serious for the Cadoux algorithm, which consists in solving a sequence of SOCQP that become more and more similar as the fixed-point algorithm converges. This motivates the investigation of algorithms with not-as-good convergence properties as interior-point methods, but whose potential slowness on single problems may be compensated by their ability to be easily warm-started.

Heyn's algorithm While we have not implemented nor tested this algorithm, we would like to mention that Heyn (2013, Section 4.5) also proposes a direct primal-dual interior-point method that does not require the SOCP reformulation. He observes very good convergence w.r.t. the number of iterations, but deplores that the computational cost per iteration is much higher than for proximal or splitting methods (which we will discuss below).

3.3 First-order proximal methods

3.3.1 Proximal operator

We refer the reader to (Parikh and S. P. Boyd 2014) for an excellent introduction and overview of proximal algorithms. In this manuscript we will focus on very narrow applications of this framework, and won't make use of the more abstract general setting. Yet, some knowledge about proximal methods may help in finding links between the different methods presented below.

Definition 3.1. *The proximal operator with coefficient $\beta>0$ of a closed proper convex function*[5] *$f:X\to\bar{\mathbb{R}}$, with X a reflexive Banach space, is defined as the solution to a convex minimization problem,*

$$\begin{aligned} &\operatorname{prox}_{\beta} f : X \to X \\ &\qquad \boldsymbol{x} \mapsto \arg\min_{\boldsymbol{z}\in X} f(\boldsymbol{z}) + \frac{1}{2\beta}\|\boldsymbol{x}-\boldsymbol{z}\|. \end{aligned} \tag{3.5}$$

For $\beta=+\infty$, $\operatorname{prox}_{\beta} f$ may be multi-valued, as any point of X optimum for f satisfies the definition. However, for $\beta<+\infty$, the function being minimized becomes strictly convex and

[5] See Appendix A.1.1

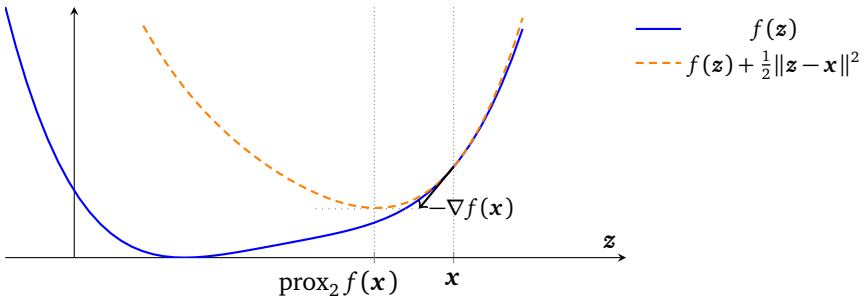


Figure 3.2: *Geometrical evaluation of the proximal operator and relationship with a gradient descent step*

coercive, and therefore from Theorem A.5, the proximal operator is uniquely defined. As β goes towards 0, the minimization problem becomes more and more skewed towards the point at which the operator is evaluated. Alternatively, as illustrated in Figure 3.2, evaluating the proximal operator $\operatorname{prox}_{\beta} f(\boldsymbol{x})$ may be seen as moving $\boldsymbol{x}$ towards the minimum of f; the lower β, the smaller the step size. More precisely, Parikh and S. P. Boyd (2014, Section 3.3) point out that when f is twice-differentiable,

$$\operatorname{prox}_{\beta} f(\boldsymbol{x}) = \boldsymbol{x} - \beta(\nabla f)(\boldsymbol{x}_0) + o(\beta). \tag{3.6}$$

The fixed-point algorithm defined by the induction $\boldsymbol{x}^{k+1} := \operatorname{prox}_{\beta} f(\boldsymbol{x}^k)$ can be thus be seen as a modified gradient descent step. Corroborating this interpretation, it can be shown that $\boldsymbol{x}_0 \in X$ is a minimum for f on X if and only if $\operatorname{prox}_{\beta} f(\boldsymbol{x}_0) = \boldsymbol{x}_0$, i.e., if and only if $\boldsymbol{x}_0$ is a fixed-point of the proximal operator (Parikh and S. P. Boyd 2014, Section 2.3).

An interesting property of the proximal operator manifests itself when $f = \mathscr{I}_C$, the characteristic function of $C \subset X$, a proper, closed, convex subset of X (Definition A.8). Then

$$\operatorname{prox}_{\beta} \mathscr{I}_C(\boldsymbol{x}) = \arg\min_{\boldsymbol{z} \in C} \frac{1}{2\beta}\|\boldsymbol{x} - \boldsymbol{z}\| = \Pi_C(\boldsymbol{x}),$$

the orthogonal projection on C. This interpretation of the orthogonal projection as a special case of a proximal operator translates to the optimality conditions of convex optimization problems. Recall (Remark A.3) that the optimality condition for the minimization problem (A.4),

$$\min_{\boldsymbol{x} \in X} f(\boldsymbol{x}) + \mathscr{I}_C(\boldsymbol{x}) \tag{A.4}$$

reads, when f is differentiable on C, and for any $\alpha \in \mathbb{R}_+^*$,

$$\Pi_C\left(\boldsymbol{x}_0 - \alpha(\nabla f)(\boldsymbol{x}_0)\right) = \boldsymbol{x}_0;$$

a similar but more general optimality condition can be stated on proximal operators (Parikh and S. P. Boyd 2014, Section 4.2).

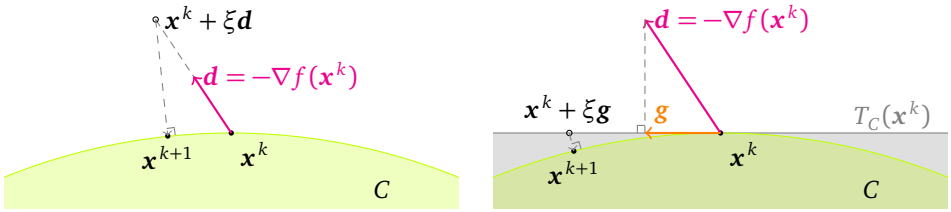
Property 3.1. *The optimality conditions of the minimization problem*

$$\min_{\boldsymbol{x} \in X} f(\boldsymbol{x}) + g(\boldsymbol{x})$$

with f and g proper closed convex and f differentiable on $\operatorname{dom} g$ *are*

$$\operatorname{prox}_{\beta} g\left(\boldsymbol{x}_0 - \alpha(\nabla f)(\boldsymbol{x}_0)\right) = \boldsymbol{x}_0, \quad \alpha \in \mathbb{R}_+^*.$$

Any fixed-point of the sequence induced by $\boldsymbol{x}^{k+1} := \operatorname{prox}_{\beta} g\left(\boldsymbol{x}^k - \alpha(\nabla f)(\boldsymbol{x}^k)\right)$ will thus yield a minimum of $f + g$ over X. When (∇f) is Lipschitz with constant L, this fixed-point algorithm will converge as $O(1/k)$ for $\beta \in]0, \frac{1}{L}]$ (Parikh and S. P. Boyd 2014) — though it is only practical as long as evaluating $\operatorname{prox}_{\beta} g$ is simple enough. The projected gradient descent method, presented below, will thus appear as a special case of fixed-point algorithms on a proximal operator where $g = \mathscr{I}_C$.



(a) *Projected gradient descent* **(b)** *Projected-gradient*

Figure 3.3: *Comparison of the projected gradient descent and projected-gradient steps at a point x^k in the boundary of the feasible set C.*

3.3.2 Projected Gradient Descent

The projected gradient descent algorithm simply consists in taking a step in the direction opposite to the objective function's gradient, then projecting the result onto the feasible set.

Applied to the dual SOCQP (2.21), it reads as Algorithm 3.1,

Algorithm 3.1: Canonical Projected Gradient Descent Algorithm

Result: r approximate solution to $\arg\min_{r \in \mathcal{K}_\mu} \frac{1}{2} r^\intercal W r + r^\intercal b$

for $k \in \mathbb{N}$ **do**

$u^k \leftarrow \tilde{W} r^k + \tilde{b}$;

Compute step size ξ^{k+1} using a line-search procedure;

3.1.4 $r^{k+1} \leftarrow \Pi_{\mathcal{K}_\mu}\left(r^k - \xi^{k+1} u^k\right)$;

end

It is easy to see that any fixed-point of this procedure will satisfy $\Pi_{\mathcal{K}_\mu}(r - \xi u) = r$, and thus will be an optimum of the dual SOCQP. The iterative process can also be terminated as soon as the norm of a complementarity function (see Section 3.1.2) at (u, r) yields an error below a chosen tolerance.

Theoretically, a similar algorithm could also be written for the primal SOCQP; however, except for a very specific structure of H, the projection on the feasible set $V := \{v \in \mathbb{R}^m, Hv = w \in \mathcal{K}_{\frac{1}{\mu}}\}$ cannot be explicitly computed.

Algorithm 3.1 can also be repurposed to directly solve the DCFP (instead of the convexified version). Indeed, as mentioned earlier, the projection gradient descent algorithm can simply be seen as a fixed-point algorithm with a varying step size. Replacing u^k in line 3.1.4 with $\tilde{u}(u^k)$, we obtain the fixed-point algorithm on the De Saxcé formulation (with a well-chosen step size); replacing $\Pi_{\mathcal{K}_\mu}$ with $\mathcal{T}_{0,\mu r_N}$, the fixed-point algorithm on the Alart–Curnier formulation. However, these algorithms no longer minimize convex functions under convex constraints, and the theoretical convergence properties are lost — which does not prevent them from sometimes being quite effective in practice, as confirmed by our benchmarks presented in Chapters 4 and 7.

Acceleration Under regularity conditions, the projected gradient descent algorithm converges as $O(1/k)$ (Parikh and S. P. Boyd 2014); however, incorporating a momentum term may yield convergence as $O(1/k^2)$. Heyn (2013) applied the Nesterov acceleration (Nesterov 1983) to SOCQP minimization, yielding the APGD method and allowing efficient solving of very large convexified DCFP (Mazhar et al. 2015).

Projected gradient When on the boundary and close to the solution, the gradient will be very close to the normal cone of the constraint, and therefore unless the step size is chosen to be quite

big, the projected position will be very close to the original position (Figure 3.3). The projected-gradient algorithm alleviates this problem by first projecting the gradient onto the tangent cone T_C to the constraint $C \subset X$,

$$T_C(\boldsymbol{x}) := \overline{\{\boldsymbol{y} \in X,\ \exists \epsilon > 0,\ \forall 0 < t < \epsilon, (\boldsymbol{x} + t\boldsymbol{y}) \in C\}}.$$

Algorithm 3.2: Projected-Gradient Algorithm

for $k \in \mathbb{N}$ **do**

$\quad \boldsymbol{u}^k \leftarrow \tilde{W}\boldsymbol{r}^k + \tilde{b}$;

3.2.3 $\quad \boldsymbol{g}^{k+1} \leftarrow \Pi_{T_{\mathscr{K}_\mu}(\boldsymbol{r}^k)}\boldsymbol{u}^k$;

$\quad$ Compute step size ξ^{k+1} using a line-search procedure;

$\quad \boldsymbol{r} \leftarrow \Pi_{\mathscr{K}_\mu}\left(\boldsymbol{r}^k - \xi^{k+1}\boldsymbol{g}^{k+1}\right)$;

end

If the projection tangent cone is hard to compute, it can be approximated by performing another projection on the feasible set C. Line 3.2.3 is then replaced with two steps,

$\boldsymbol{x} \leftarrow \Pi_{\mathscr{K}_\mu}(\boldsymbol{r} - \varrho\boldsymbol{g})$;

$\boldsymbol{g} \leftarrow \boldsymbol{x} - \boldsymbol{r}$;

Other variants Many other variants of the projected-gradient and projected gradient descent exist. We can mention the Spectral Projected-Gradient (SPG) method, with was observed to work well on convexified DCFP by Tasora (2013). In Appendix B.2, we propose a Nesterov-accelerated, line-search free variant of this algorithm, which we coined **ASPG** and that performed consistently well on our problems (both convexified and standard). Algorithm B.2 is implemented in the `bogus` library, along with the APGD method, canonical algorithms, and a projected-gradient variant augmented with a simple conjugation step.

3.3.3 Primal–dual proximal methods

Primal–dual proximal methods are a class of first-order minimization algorithms that work by splitting the objective function, and iterating on both the primal variable (in our case, the velocity $\boldsymbol{v}$) and the dual variable (the forces $\boldsymbol{r}$). The methods presented below were not observed to converge very well on our DCFP without requiring significant manual tuning of the step size parameters, and we therefore won't use them in practice. However, their popularity in computational fluid mechanics make them worthy of interest.

Alternating Directions Method of Multipliers ADMM is another way of minimizing separable problems, introduced by Glowinski and Marroco (1975) for non-differentiable continuum flow problems and made popular by Fortin and Glowinski (1983) under the name of Augmented Lagrangian algorithm. ADMM has also been recently introduced to the Computer Graphics simulation community by Narain, Overby, et al. (2016), though in a slightly different setting.

The minimization of $f(\boldsymbol{v}) + g(H\boldsymbol{v} + \boldsymbol{w})$ is written as

$$\min_{\boldsymbol{u}=H\boldsymbol{v}+\boldsymbol{w}} f(\boldsymbol{v}) + g(\boldsymbol{u})$$

The Lagrangian associated to this minimization problem is

$$\mathscr{L} = f(\boldsymbol{v}) + g(\boldsymbol{u}) + \boldsymbol{r}^\intercal(\boldsymbol{u} - H\boldsymbol{v} - \boldsymbol{w}),$$

and the augmented Lagrangian with parameter γ is

$$\mathscr{L}_\gamma = f(\boldsymbol{v}) + g(\boldsymbol{u}) + \boldsymbol{r}^\intercal(\boldsymbol{u} - H\boldsymbol{v} - \boldsymbol{w}) + \frac{\gamma}{2}\|H\boldsymbol{v} + \boldsymbol{w} - \boldsymbol{u}\|^2.$$

We are looking for a saddle-point of $\mathscr{L}_\gamma$. Minimizing over the first two variables $\boldsymbol{v}$ and $\boldsymbol{u}$ in turn, and taking a gradient step to maximize $\boldsymbol{\lambda}$, we obtain Algorithm 3.3,

Algorithm 3.3: Alternating Directions Method of Multipliers

for $k \in \mathbb{N}$ **do**

3.3.2 $\quad \boldsymbol{v}^{k+1} \leftarrow \arg\min_{\boldsymbol{v}} \mathscr{L}_\gamma(\boldsymbol{v}, \boldsymbol{u}^k, \boldsymbol{r}^k)$;

3.3.3 $\quad \boldsymbol{u}^{k+1} \leftarrow \arg\min_{\boldsymbol{u}} \mathscr{L}_\gamma(\boldsymbol{v}^{k+1}, \boldsymbol{u}, \boldsymbol{r}^k)(\boldsymbol{v}^{k+1} + \boldsymbol{r}k)$;

$\quad \boldsymbol{r}^{k+1} \leftarrow \boldsymbol{r}^k - \gamma(H\boldsymbol{v} + \boldsymbol{w} - \boldsymbol{u})$;

end

Now, Line 3.3.3 can easily be written as a proximal operator:

$$\begin{aligned}\boldsymbol{u}^{k+1} &:= \arg\min_{\boldsymbol{u}} g(\boldsymbol{u}) + \frac{\gamma}{2}\|H\boldsymbol{v}^{k+1} + \boldsymbol{w} - \boldsymbol{u} - \frac{\boldsymbol{r}^k}{\gamma}\| \\ &= \operatorname{prox}_{\frac{1}{\gamma}} g(H\boldsymbol{v}^{k+1} + \boldsymbol{w} - \frac{1}{\gamma}\boldsymbol{r}^k)\end{aligned}$$

The first step, Line 3.3.2, is more tricky, but can be dealt with by linearizing the quadratic term around $\boldsymbol{v}^k$ and adding a new quadratic regularization term, (Parikh and S. P. Boyd 2014, Section 4.4.2),

$$\begin{aligned}\frac{\gamma}{2}\|H\boldsymbol{v} + \boldsymbol{w} - \boldsymbol{u}^k\|^2 &\sim \gamma\left\langle H\boldsymbol{v}^k + \boldsymbol{w} - \boldsymbol{u}^k, H\boldsymbol{v}\right\rangle + \frac{\beta}{2}\|\boldsymbol{v} - \boldsymbol{v}^k\| \\ &\sim \frac{\beta}{2}\|\boldsymbol{v} - \boldsymbol{v}^k + \frac{\gamma}{\beta}H^\intercal\left(H\boldsymbol{v}^k + \boldsymbol{w} - \boldsymbol{u}^k\right)\|^2.\end{aligned}$$

This yields the *linearized* ADMM algorithm, with Line 3.3.2 replaced by

$$\begin{aligned}\boldsymbol{v}^{k+1} &\leftarrow \arg\min_{\boldsymbol{v}} f(\boldsymbol{v}) + \frac{\beta}{2}\|\boldsymbol{v} - \boldsymbol{v}^k + \frac{\gamma}{\beta}H^\intercal\left(H\boldsymbol{v}^k + \boldsymbol{w} - \boldsymbol{u}^k\right) - \frac{1}{\beta}H^\intercal\boldsymbol{r}^k\|^2 \\ &= \operatorname{prox}_{\frac{1}{\beta}} f\left(\boldsymbol{v}^k + \frac{1}{\beta}H^\intercal\boldsymbol{r}^k - \frac{\gamma}{\beta}H^\intercal\left(H\boldsymbol{v}^k + \boldsymbol{w} - \boldsymbol{u}^k\right)\right) \\ &= \operatorname{prox}_{\frac{1}{\beta}} f\left(\boldsymbol{v}^k + \frac{1}{\beta}H^\intercal\left(\boldsymbol{r}^k - \gamma\left(H\boldsymbol{v}^k + \boldsymbol{w} - \boldsymbol{u}^k\right)\right)\right)\end{aligned} \tag{3.7}$$

The ADMM algorithm 3.3 can also be interpreted as an integral controller, where $\boldsymbol{r}$ is the control $\boldsymbol{u}$ the state, and the goal is to achieve $\boldsymbol{u} = H\boldsymbol{v} + \boldsymbol{w}$ (Parikh and S. P. Boyd 2014).

ADMM on primal formulation Now, let us consider our primal SOCQP (without linear constraints for now). We have $f(\boldsymbol{v}) = \frac{1}{2}\boldsymbol{v}^\intercal M\boldsymbol{v} - \boldsymbol{v}^\intercal\boldsymbol{f}$, and $g(\boldsymbol{u}) := \mathscr{I}_{\mathscr{K}_{\frac{1}{\mu}}}(\boldsymbol{u})$. Therefore

$$\begin{aligned}\operatorname{prox}_{\frac{1}{\beta}} f(\boldsymbol{x}) &= \arg\min_{\boldsymbol{v}} \frac{1}{2}\boldsymbol{v}^\intercal M\boldsymbol{v} - \boldsymbol{v}^\intercal\boldsymbol{f} + \frac{\beta}{2}\|\boldsymbol{v} - \boldsymbol{x}\|^2 \\ &= \arg\min_{\boldsymbol{v}} \frac{1}{2}\boldsymbol{v}^\intercal(M + \beta\mathbb{I})\boldsymbol{v} - \boldsymbol{v}^\intercal(\boldsymbol{f} + \beta\boldsymbol{x}) \\ &= (M + \beta\mathbb{I})^{-1}(\boldsymbol{f} + \beta\boldsymbol{x}) \\ \operatorname{prox}_{\frac{1}{\gamma}} g(\boldsymbol{y}) &= \Pi_{\mathscr{K}_{\frac{1}{\mu}}}(\boldsymbol{y}).\end{aligned}$$

The linearized ADMM algorithm on the primal SOCQP thus becomes Algorithm 3.4,

Algorithm 3.4: Linearized ADMM for SOCQP

for $k \in \mathbb{N}$ **do**

$\boldsymbol{v}^{k+1} \leftarrow (M+\beta\mathbb{I})^{-1}\left(\boldsymbol{f}+\beta\boldsymbol{v}^k+H^\intercal\left(\boldsymbol{r}^k-\gamma\left(H\boldsymbol{v}^k+\boldsymbol{w}-\boldsymbol{u}^k\right)\right)\right)$;

$\boldsymbol{u}^{k+1} \leftarrow \Pi_{\mathscr{K}_{\frac{1}{\mu}}}\left(H\boldsymbol{v}^{k+1}+\boldsymbol{w}-\frac{1}{\gamma}\boldsymbol{r}^k\right)$;

$\boldsymbol{r}^{k+1} \leftarrow \boldsymbol{r}^k-\gamma(H\boldsymbol{v}^{k+1}+\boldsymbol{w}-\boldsymbol{u}^{k+1})$;

end

We can easily verify that any fixed point $(\boldsymbol{v},\boldsymbol{u},\boldsymbol{r})$ of this algorithm satisfies

$$\begin{cases} M\boldsymbol{v}=\boldsymbol{f}+H^\intercal\boldsymbol{r} \\ \boldsymbol{u}=H\boldsymbol{v}+\boldsymbol{w} \\ \boldsymbol{r}\in-\mathscr{N}_{\mathscr{K}_{\frac{1}{\mu}}}\boldsymbol{u} \end{cases}$$

and is thus a solution of the primal SOCQP without linear constraints.

The Alternating Minimization Algorithm (AMA, see Goldstein et al. 2014) is obtained by suppressing the quadratic term in the expression of the Lagrangian when minimizing w.r.t. $\boldsymbol{v}$, that is, replacing Line 3.3.2 with

$$\boldsymbol{v}^{k+1} \leftarrow \arg\min_{\boldsymbol{v}} f(\boldsymbol{v})-\left\langle \boldsymbol{r}^k, H\boldsymbol{v}\right\rangle$$

Applied to our SOCQP, we obtain Algorithm 3.5,

Algorithm 3.5: AMA for SOCQP

for $k \in \mathbb{N}$ **do**

$\boldsymbol{v}^{k+1} \leftarrow M^{-1}(\boldsymbol{f}+H^\intercal\boldsymbol{r}^k)$;

$\boldsymbol{u}^{k+1} \leftarrow \Pi_{\mathscr{K}_{\frac{1}{\mu}}}\left(H\boldsymbol{v}^{k+1}+\boldsymbol{w}-\frac{1}{\gamma}\boldsymbol{r}^k\right)$;

$\boldsymbol{r}^{k+1} \leftarrow \boldsymbol{r}^k-\gamma(H\boldsymbol{v}^{k+1}+\boldsymbol{w}-\boldsymbol{u}^{k+1})$;

end

In contrast to the original ADMM algorithm, AMA may be ill-defined when M is only positive semi-definite. Both algorithms can be easily extended to handle linear constraints; for ADMM, we get Algorithm 3.6,

Algorithm 3.6: Linearized ADMM for SOCQP with linear constraints

for $k \in \mathbb{N}$ **do**

$\boldsymbol{v}^{k+1} \leftarrow$
$(M+\beta\mathbb{I})^{-1}\left(\boldsymbol{f}+\beta\boldsymbol{v}^k+H^\intercal\left(\boldsymbol{r}^k-\gamma_1\left(H\boldsymbol{v}^k+\boldsymbol{w}-\boldsymbol{u}^k\right)\right)+C^\intercal\left(\boldsymbol{\lambda}^k-\gamma_2\left(C\boldsymbol{v}^k-\boldsymbol{k}\right)\right)\right)$;

$\boldsymbol{u}^{k+1} \leftarrow \Pi_{\mathscr{K}_{\frac{1}{\mu}}}\left(H\boldsymbol{v}^{k+1}+\boldsymbol{w}-\frac{1}{\gamma_1}\boldsymbol{r}^k\right)$;

$\boldsymbol{r}^{k+1} \leftarrow \boldsymbol{r}^k-\gamma_1(H\boldsymbol{v}^{k+1}+\boldsymbol{w}-\boldsymbol{u}^{k+1})$;

$\boldsymbol{\lambda}^{k+1} \leftarrow \boldsymbol{\lambda}^k-\gamma_2(C\boldsymbol{v}^{k+1}-\boldsymbol{k})$;

end

Like the projected-gradient algorithm on the dual SOCQP, this algorithm requires solving at each iteration a linear systems with M (or $M+\beta\mathbb{I}$) as left-hand-side. However, the constraint projection is here done on $\mathscr{K}_{\frac{1}{\mu}}$, instead of $\mathscr{K}_\mu$ for the projected gradient. One may also attempt to directly solve the original DCFP by replacing the term $\boldsymbol{w}$ in each step with the expression $\boldsymbol{w}+\boldsymbol{s}(H\boldsymbol{v}+\boldsymbol{w})$; however, once again the theoretical convergence properties are then lost.

Starting from dual SOCQP we can also define an AMA algorithm that will only require one multiplication by M at each iteration.

AMA on dual formulation We write the dual SOCQP (2.21) without linear constraints as

$$\min_{z=H^\intercal r} \underbrace{\frac{1}{2} z^\intercal M^{-1} z + z^\intercal M^{-1} f}_{f(z)} + \underbrace{r^\intercal w + \mathscr{I}_{\mathscr{K}_\mu}(r)}_{g(r)}.$$

Writing the AMA on this formulation yields Algorithm 3.7,

Algorithm 3.7: AMA on dual formulation

for $k \in \mathbb{N}$ **do**

$z^{k+1} \leftarrow \arg\min_z f(z) - \langle z, v^k \rangle$;

3.7.3 $r^{k+1} \leftarrow \arg\min_r g(r) + \langle H^\intercal r, v^k \rangle + \frac{\gamma}{2}\|z^{k+1} - H^\intercal r\|^2$;

$v^{k+1} \leftarrow v^k + \gamma(H^\intercal r - z)$;

end

The first minimization is straightforward;

$$\begin{aligned} z^{k+1} &= \arg\min_z \frac{1}{2} z^T M^{-1} z + z^\intercal \left(M^{-1} f - v^k\right) \\ &= M(v^k - f). \end{aligned}$$

However the second step has to be reworked. Linearizing once again the quadratic part, we replace Line 3.7.3 with

$$\begin{aligned} r^{k+1} &\leftarrow \arg\min_r g(r) + \langle H^\intercal r, v^k \rangle + \gamma \langle H^\intercal r^k - z, H^\intercal r \rangle + \frac{\beta}{2}\|r - r^k\|^2 \\ &= \arg\min_r g(r) + \frac{\beta}{2}\|r - r^k + \frac{1}{\beta} H\left(v^k + \gamma\left(H^\intercal r^k - z\right)\right)\|^2 \\ &= \arg\min_{r \in \mathscr{K}_\mu} \frac{\beta}{2}\|r - r^k + \frac{1}{\beta} w + \frac{1}{\beta} H\left(v^k + \gamma\left(H^\intercal r^k - z^{k+1}\right)\right)\|^2 \\ &= \Pi_{\mathscr{K}_\mu}\left(r^k - \frac{1}{\beta}\left[w + H\left(v^k + \gamma\left(H^\intercal r^k - z\right)\right)\right]\right). \end{aligned}$$

We see easily that any fixed-point (v, z, r) of this linearized version of Algorithm 3.7 satisfies

$$\begin{cases} Mv = z + f \\ z = H^\intercal r \\ Hv + w \in -\mathscr{N}_{\mathscr{K}_\mu}(r), \end{cases}$$

and thus is also a solution of the convexified DCFP. Once again, this algorithm can be extended to handle linear constraints, by driving the auxiliary variable z to satisfy $z = H^\intercal r + C^\intercal \lambda$. We obtain Algorithm 3.8,

Algorithm 3.8: Linearized AMA on dual formulation with linear constraints

for $k \in \mathbb{N}$ **do**

$z^{k+1} \leftarrow Mv^k + f$;

3.8.3 $r^{k+1} \leftarrow \Pi_{\mathscr{K}_\mu}\left(r^k - \frac{1}{\beta_1}\left[w + H\left(v^k + \gamma\left(H^\intercal r^k + C^\intercal \lambda^k - z^{k+1}\right)\right)\right]\right)$;

$\lambda^{k+1} \leftarrow \lambda^k - \frac{1}{\beta_2}\left[-k + C\left(v^k + \gamma\left(H^\intercal r^k + C^\intercal \lambda^k - z^{k+1}\right)\right)\right]$;

$v^{k+1} \leftarrow v^k + \gamma(H^\intercal r^{k+1} + C^\intercal \lambda^{k+1} - z^{k+1})$;

end

Just like with the primal formulation, one can directly solve the original DCFP by replacing w on line 3.8.3 with $w + s(Hv + w)$.

Variants Goldstein et al. 2014 also define Nesterov-like accelerated versions of those algorithms. Other variants of such separation schemes exist, such as the Arrow–Hurwitz algorithm (which is itself an extension the Uzawa algorithm to the proximal setting):

---

for $k \in \mathbb{N}$ **do**
  $\boldsymbol{r}^{k+1} \leftarrow$ -prox$_{\frac{1}{\gamma}}\, g^\star\left(-\boldsymbol{r}^k + \gamma\left(H\boldsymbol{v}^k + \boldsymbol{w}\right)\right)$;
  $\boldsymbol{v}^{k+1} \leftarrow \text{prox}_{\frac{1}{\beta}}\, f(\boldsymbol{v}^k + \frac{1}{\beta}H^\intercal \boldsymbol{r}^{k+1})$;
end

---

which for our primal SOCQP is written

Algorithm: Arrow–Hurwitz algorithm for SOCQP

for $k \in \mathbb{N}$ **do**
  $\boldsymbol{r}^{k+1} \leftarrow \Pi_{\mathscr{K}_\mu}\left(\boldsymbol{r}^k - \gamma\left(H\boldsymbol{v}^k + \boldsymbol{w}\right)\right)$;
  $\boldsymbol{v}^{k+1} \leftarrow (M + \beta\mathbb{I})^{-1}(\boldsymbol{f} + \beta\boldsymbol{v}^k + H^\intercal \boldsymbol{r}^{k+1})$;
end

Compared to our original ADMM algorithm, the constraint projection is done on the forces $\boldsymbol{r}$ instead of on the relative velocities $\boldsymbol{u}$; the Arrow-Hurwitz degrades to a projected gradient descent algorithm (without line search) when $\beta = 0$. The popular Chambolle and Pock (2010) algorithm is an accelerated version of the Arrow–Hurwitz algorihm.

The `bogus` library (Daviet 2013), implements the accelerated ADMM from (Goldstein et al. 2014) and a preconditioned and accelerated version of the AMA on the dual SOCQP. In practice, our early tests were not very satisfying; fine-tuning of the coefficients β and γ was require to achieve fast convergence.

3.4 Splitting methods

3.4.1 Operator splitting

The most popular way to solve the dual formulation of the DCFP (2.19) is to use a splitting method: that is, treating one part of the operator W implicitly, and another part explicitly. In practice, this means decomposing W as $W = W_1 + W_2$, and iterating as

$$\boldsymbol{u}^{k+1} = W_1\boldsymbol{r}^{k+1} + W_2\boldsymbol{r}^k + C^\intercal\boldsymbol{\lambda}^k + \tilde{b} \tag{3.8}$$

$$(\boldsymbol{u}_i^{k+1}, \boldsymbol{r}_i^{k+1}) \in \mathscr{C}_{\mu_i} \forall 1 \leq i \leq n. \tag{3.9}$$

$$P\boldsymbol{\lambda}^{k+1} = \boldsymbol{c} - B^\intercal\boldsymbol{r}^{k+1}. \tag{3.10}$$

a fixed-point for $(\boldsymbol{r}, \boldsymbol{\lambda})$ is reached, that is $(\boldsymbol{r}^{k+1}, \boldsymbol{\lambda}^{k+1}) = (\boldsymbol{r}^k, \boldsymbol{\lambda}^k)$.

Obviously, W_1 and W_2 should be chosen so that the resulting problem (3.8 – 3.9) is easier to solve than the original DCFP. The convexified DCFP can also be solved in this manner, replacing the Coulomb law condition (3.9) with $\mathscr{K}_{\frac{1}{\mu_i}} \ni \boldsymbol{u}_i^{k+1} \perp \boldsymbol{r}_i^{k+1} \in \mathscr{K}_{\mu_i}$.

In practice, W has a sparse block structure composed of $d \times d$ blocks (W_{OJ}), $1 \leq i, j \leq n$, where each block (W_{ij}) describe the interaction between the i^{th} and j^{th} contacts (that is, the relative velocity change at i when a force increment is applied at j). W can thus be decomposed as the sum of a block-diagonal matrix $W_D := (W_{ii})$, a lower part $W_L := (W_{i,j})_{i>j}$ and an upper part $W_U := (W_{i,j})_{i<j}$.

Choosing $W_1 = D$ and $W_2 = L + U$ yields the Jacobi algorithm; equation (3.8) becomes separable, and result in n d-dimensional problems that can be solved in parallel:

$$\begin{cases} \boldsymbol{u}_i^{k+1} = W_{ii}\boldsymbol{r}_i^{k+1} + \sum_{j\neq i} W_{i,j}\boldsymbol{r}_j^k + C^\intercal\boldsymbol{\lambda}^k + \tilde{b} \\ \qquad (\boldsymbol{u}_i^{k+1}, \boldsymbol{r}_i^{k+1}) \in \mathscr{C}_{\mu_i}. \end{cases}$$

Choosing $W_1 = D + L$ and $W_2 = U$, we obtain the Gauss–Seidel algorithm, which boasts better convergence but loses parallelism; the d-dimensional local problems now have to be solved in order:

$$\begin{cases} \boldsymbol{u}_i^{k+1} = W_{ii}\boldsymbol{r}_i^{k+1} + \sum_{j<i} W_{i,j}\boldsymbol{r}_j^{k+1} + \sum_{j>i} W_{i,j}\boldsymbol{r}_j^{k} + C^\intercal \boldsymbol{\lambda}^k + \tilde{b} \\ \qquad (\boldsymbol{u}_i^{k+1}, \boldsymbol{r}_i^{k+1}) \in \mathscr{C}_{\mu_i}. \end{cases}$$

In both cases, solving our global problem can thus be reduced to solving a sequence of local problems (or *one-contact* problems),

$$\begin{cases} \boldsymbol{u}_i = W_{ii}\boldsymbol{r}_i + \bar{\boldsymbol{b}}_i \\ (\boldsymbol{u}_i, \boldsymbol{r}_i) \in \mathscr{C}_{\mu_i}, \end{cases} \tag{3.11}$$

where W_{ii} is a $d \times d$, positive semi-definite matrix.

This one-contact problem is much easier to solve than the original one; indeed, for $d = 2$ or $d = 3$, we can construct an analytical solver, as we will show in Chapter 4.

Splitting methods have been used on contact-related problems for several decades. Raous et al. (1988) applied them to the statics of an elastic body under frictional contact, and mention previous uses in the frictionless case. However, their huge popularity nowadays is largely due to the advent of the *Nonsmooth Contact Dynamics* (NSCD) method (Jean 1999; Jean and Moreau 1992; Jourdan et al. 1998), which proposed to compute the dynamics of Discrete Element Models with frictional contacts using the Gauss–Seidel algorithm and an efficient one-contact solver.

3.4.2 Convergence properties

Interpretation as block-coordinate descent When solving the convexified DCFP, the local problem (3.11) becomes

$$\begin{cases} \boldsymbol{u}_i = W_{ii}\boldsymbol{r}_i + \bar{\boldsymbol{b}}_i \\ \boldsymbol{u}_i \in -\mathscr{N}_{\mathscr{K}_{\mu_i}}(\boldsymbol{r}_i), \end{cases} \tag{3.12}$$

which from Theorem A.6 we recognize as the optimality conditions of the minimization problem

$$\operatorname*{arg\,min}_{\boldsymbol{r}_i \in \mathscr{K}_{\mu_i}} \frac{1}{2}\boldsymbol{r}_i^\intercal W_{ii}\boldsymbol{r}_i + \boldsymbol{r}_i^\intercal \bar{\boldsymbol{b}}_i = \operatorname*{arg\,min}_{\boldsymbol{r}_i \in \mathscr{K}_{\mu_i}} J_i(\boldsymbol{r}_i), \tag{3.13}$$

where $J_i(\boldsymbol{r}_i) := \frac{1}{2}\boldsymbol{r}^\intercal W \boldsymbol{r} + \boldsymbol{r}^\intercal \boldsymbol{b}$ is defined by freezing every component of $\boldsymbol{r}$ except for $\boldsymbol{r}_i$. As such, successively solving the local problems (3.12) amounts to iteratively minimizing the quadratic objective function $J(\boldsymbol{r}) := \frac{1}{2}\boldsymbol{r}^\intercal W \boldsymbol{r} + \boldsymbol{r}^\intercal \boldsymbol{b}$ w.r.t. each contact force $\boldsymbol{r}_i$; the Gauss–Seidel algorithm can thus be seen as a block-coordinate descent method.

If W_{ii} is symmetric, positive-definite for all i, then each local problem admits a unique solution, and the objective function will monotonically decrease to its optimal value (Grippo and Sciandrone 2000).

Proximal modification However, W_{ii} may only be positive semi-definite, and in this case the minimization problem (3.13) will be ill-posed. However, we can modify it slightly as

$$\operatorname*{arg\,min}_{\boldsymbol{r}_i \in \mathscr{K}_{\mu_i}} \frac{1}{2}\boldsymbol{r}^\intercal W \boldsymbol{r} + \boldsymbol{r}^\intercal \boldsymbol{b} + \frac{\beta}{2}\|\boldsymbol{r}_i^k - \boldsymbol{r}_i\|^2 = \operatorname{prox}_{\frac{1}{\beta}} J_i(\boldsymbol{r}_i^k)$$

with $\beta > 0$. Any fixed-point of the proximal operator will be optimal for J_i, and therefore a fixed-point of the modified Gauss–Seidel algorithm will still be a solution of the dual SOCQP. Grippo and Sciandrone (2000) show that this fixed-point will always be reached for W symmetric positive semi-definite.

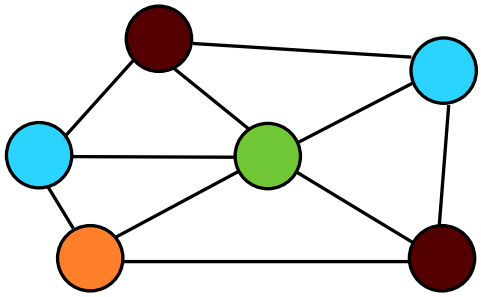


Figure 3.4: *Sample coloring of a contact graph; contacts (nodes) with the same color can be solved in parallel. Edges represent non-zero $d \times d$ blocks of the Delassus operator.*

By analogy, we define a proximal version of our splitting algorithms for the DCFP by replacing the local problem (3.11) with (3.14),

$$\begin{cases} \boldsymbol{u}_i = (W_{ii} + \beta \mathbb{I})\boldsymbol{r}_i + \bar{\boldsymbol{b}}_i - \beta \boldsymbol{r}_i^k \\ (\boldsymbol{u}_i, \boldsymbol{r}_i) \in \mathscr{C}_{\mu_i}. \end{cases} \tag{3.14}$$

However, note that when using the Gauss–Seidel to solve the DCFP (rather than the convexified DCFP), convergence is no longer guaranteed; indeed, the algorithm may even exhibit cycles. Jourdan et al. (1998) still manage to derive a convergence proof for a particular 2D scenario.

3.4.3 Performance

Accelerations The Jacobi and Gauss–Seidel names come from the analogy with similar algorithms devised to solve linear systems, and just like their siblings, the splitting methods applied to the DCFP feature a quite slow asymptotic convergence.

Instead of using directly the solution of the local problem (3.11) as $\boldsymbol{r}_i^{k+1}$, one may store this solution as $\boldsymbol{r}_i^{k+\frac{1}{2}}$, and compute the final iterate $\boldsymbol{r}_i^{k+1}$ as $\boldsymbol{r}_i^{k+1} = (1-\omega)\boldsymbol{r}_i^k + \omega \boldsymbol{r}_i^{k+1}$, with ω a positive coefficient. For $\omega = 1$ we retrieve the Gauss–Seidel algorithm, and for $\omega > 1$ the resulting method is known as the Successive Over-Relaxation (SOR) algorithm. When used to solve linear systems with a symmetric positive-definite matrix, SOR has been proved to converge for $0 < \omega < 2$ (Saad 2003, Theorem 4.6). However, finding an ω that will improve the convergence rate is in general a hard problem. Raous et al. (1988) proposed to accelerate convergence using the Aitken extrapolation (also known as the Δ^2 rule), which can also be interpreted as a SOR method with a well-chosen ω. Another way to potentially accelerate the convergence is the Symmetric-SOR method, which alternates between solving the local problems from $i = 1$ to n and from n to 1. The contact force updates are then propagated back-and-forth inside the contact chains, instead of in only one direction.

However, we did not find any of these accelerations to perform robustly in practice.

Parallelism While the Jacobi algorithm is embarrassingly parallel, the Gauss–Seidel is inherently sequential. On the modern many-cores architectures, this can be a significant drawback.

We can define the adirectional contact graph G, $G := (V, E)$ with vertices $V = \{1 \dots n\}$ and edges $E = \{1 \leq i < j \leq n, W_{ij} \neq \mathbf{0}\}$. Updating the force $\boldsymbol{r}_i$ will influence the right-hand-side $\bar{\boldsymbol{b}}_j$ if and only if $(i, j) \in E$. As such, every disjoint subgraph of G can be solved in parallel using the Gauss–Seidel algorithm. However, in practice it is common to have a fully connected contact graph. A substitute solution is to use a graph partitioning software, such as the open-source SCOTCH (Pellegrini and Roman 1996), to compute a set of subgraphs G_i that are weakly connected. The contact subgraphs can then be solved for in parallel, and coupled together with

an outer Jacobi algorithm. Hopefully, the loss of convergence speed will not be too dramatic; however, the more contacts subgraphs are being used, the more degraded the convergence, as the algorithm ultimately becomes a pure Jacobi. Another solution, implemented in `bogus`, is to use a coloring approach, illustrated in Figure 3.4. When the contact graph has a grid-like structure, it is possible to color each node with a checkerboard-like pattern. Then at each Gauss–Seidel iteration, all the white nodes can be solved in parallel, then all the black nodes can be solved in parallel; this is the so-called *red-black* Gauss–Seidel algorithm. It is possible to generalize this approach to arbitrary contact graphs using more colors; however, computing an optimal coloring is a NP-hard task. A greedy algorithm can be easily implemented: each contact is considered in turn, and is either: (i) assigned one of the existing colors if it has no edge to any of the contacts in this color, or (ii) assigned a new color. However, such an algorithm will often lead to a lot of colors with very few contacts, making the overall algorithm very sequential once again. Moreover, computing the partitioning or coloring can be costly, and contacts have then to be reordered to preserve good memory locality; the gain from the recovered parallelism may not suffice to outbalance the cost of the preprocessing cost.

Another approach has been explored by Tonge et al. (2012). Each pair of contacting bodies is considered independently of the other ones, resulting in n_B distinct velocity vectors for an object in contact with n_B bodies. The n_B velocity vectors are prescribed to remain consistent using fixed-joint constraints. The algorithm then iterates between solving for each contacting pair (which can been done in parallel) and enforcing the fixed-joint constraints (for which they provide an analytical formula). This results in a very GPU-friendly method, allowing the authors to solve systems with several thousands of contacts in real-time.

Overall, we found that the most efficient solution for a number of cores that is very small w.r.t. the number of contacts is the most naive one: the set of all contacts is statically assigned to a small number of threads, and each thread is allowed to process its contacts without any synchronization with the others. This might lead to race conditions when one thread is writing a given contact force at the same time as another one is reading it; the reading thread may get back the first components of the force at iteration $k+1$, and the last components at iteration k[6]. Yet we have not found this to be a problem in practice, and observed instead a good scaling w.r.t. the number of threads. Actually, the randomness induced by the scheduling of the threads was found to sometimes *improve* the convergence of the global algorithm. The major downside, due to this randomness, is that the algorithm is no longer deterministic, and may produce significantly different results each time that it is run on a given DCFP (using the proximal version somewhat reduces this variance).

Delassus operator A fundamental requisite to achieve satisfying performance is the ability to efficiently evaluate the local problem's right-hand-side $\bar{\boldsymbol{b}}_i$, which means multiplying d rows of the Delassus operator W with the current aggregate force vector $\boldsymbol{r}$. Using row-major block-compressed-storage (BSR) for W yields good performance, but requires explicitly assembling and storing W; as mentioned in Section 2.3.1, this can be costly both in time and memory, and prohibitive if the inverse mass matrix M^{-1} is dense. Moreover, even when M is the identity matrix, $HH^\intercal$ may be much more expensive to store than H.

On the other hand, computing the i^{th} block-row–vector product $W_{i[\cdot]}\boldsymbol{r} = (HM^{-1}H^\intercal)_{i[\cdot]}\boldsymbol{r}$ analytically for each contact would be very light on memory, but incur a huge runtime cost. A good middle ground, when M is block-diagonal, is to consider an intermediate vector $\boldsymbol{z} \in \mathbb{R}^m$ storing at each-instant the matrix–vector product $M^{-1}H^\intercal\boldsymbol{r}$. In this case, $M^{-1}H^\intercal$ possesses the same block structure as H^T, and can be precomputed. Updating $\boldsymbol{z}$ when a contact force $\boldsymbol{r}_i$ changes then becomes very cheap. Indeed, it simply amounts to d scale–add operations on the columns of $M^{-1}H^\intercal$. As the rows of H (and therefore the columns of $H^\intercal$) are usually quite sparse, this operation is inexpensive. Then, the right-hand-side $\bar{\boldsymbol{b}}_i$ can be deduced from $\boldsymbol{z}$ by once again

[6]Note that reading and writing aligned double-precision floats is usually atomic on modern architectures, so this approach will not be subject to intra-component corruption.

a cheap multiplication with d rows of H. Overall, the inner loop of the Gauss–Seidel algorithm becomes:

for $1 \leq i \leq n$ **do**
  $\bar{\boldsymbol{b}}_i \leftarrow \boldsymbol{b}_i + H_{i[\cdot]}\boldsymbol{z} - W_{ii}\boldsymbol{r}_i^k$;
  Set $\boldsymbol{r}_i^{k+1}$ as solution of local problem (3.11) ;
  $\boldsymbol{z} \leftarrow \boldsymbol{z} + (M^{-1}H^\intercal)_{[\cdot]i}(\boldsymbol{r}_i^{k+1} - \boldsymbol{r}_i^k)$;
end

where only the diagonal blocks W_{ii} have now to be precomputed.

When H is stored using BSR, and when its rows are quite-sparse, this algorithm remains very efficient, and is also implemented in the `bogus` library. It is however more prone to data races when synchronization-free parallelism is used; for this reason, we periodically recompute $\boldsymbol{z}$ from scratch.

Whether one or the other of these two algorithms performs best when factoring into account the cost of computing W into account is application-dependent. In practice we usually explicitly compute W for Discrete Element Models (or generally systems with a high number of degrees of freedom w.r.t. the number of contacts), and use the matrix-free variation for continuum models (the second part of this dissertation). Note that Kaufman, Tamstorf, et al. (2014) recommends avoiding computing explicitly the Delassus operator for non-linear DEM; this seems sensible, as a sequence of DCFP with varying W but improving initial guesses has then to be solved, which mean relatively less work for the solver w.r.t. the matrix assembly time.

3.4.4 Discussion

On the one hand, despite all the above theoretical drawbacks (slow or lack of convergence, sequentiality, computation of W), the Gauss–Seidel algorithm actually performs very well in practice. This will be confirmed by the results presented over the course of the next chapters, such as in Section 4.2.3. As long as the local contact problems can be solved reliably, the global algorithm is very robust, and quickly lowers the error below a tolerance that is acceptable for a lot of purposes.

Jean (1999) states that the local problems can usually be solved using 3 or less iterations of the Newton method on the Alart–Curnier function. We have not found this to be always the case, especially on reduced-coordinates flexible models, and have observed that failure of convergence in the local solver may quickly escalate to divergence of the global solver. This has motivated the conception of a very robust local solver, to which is dedicated the following chapter.

On the other hand, the Gauss–Seidel algorithm is not adapted for problems where the inverse M^{-1} of the DCFP matrix is dense, or for massively parallel architectures. In this case, we advocate instead using proximal methods. The following chapters will show that one variant of the projected gradient descent, APSG (Algorithm B.2), performs generally well. The Dual AMA (Algorithm 3.8) is also worthy of interest as it does not require explicitly solving any linear system. However, achieving satisfying practical performance would first require devising better heuristics for choosing the step sizes.

4 A Robust Gauss–Seidel Solver and its Applications

The first section of this chapter present a few modifications over the canonical Gauss–Seidel algorithm allowing to tackle problems which cause divergence of most standard implementations. The resulting algorithm performed well on a wide range of problems, was often faster than proximal or interior-points methods, and was faster and more robust than standard GS approaches.

The last three section are dedicated to applications of this algorithm; the first one, realistic simulation hair dynamics, being the one that motivated building this solver in the first place.

Our hybrid solver (Section 4.1) was originally published in (Daviet, Bertails-Descoubes, and Boissieux 2011), in the context of hair dynamics (Section 4.3), then leveraged for the inverse design of hair (Section 4.4), and, more recently, for cloth dynamics (Section 4.3).

4.1 Hybrid Gauss–Seidel algorithm

As mentioned earlier, our applications — especially hair dynamics with reduced coordinate models (see Section 4.2) — caused regular failures of the local solvers commonly in use in the literature: Newton methods or fixed-point iterations over the Alart–Curnier or De Saxcé complementarity functions. These local failures could then escalate to global divergence of the frictional contact solver, leading to visible popping or complete blowup of the simulation.

Our first objective was thus to devise a very robust solver for the local problem (3.11).

4.1.1 SOC Fischer-Burmeister function

One first idea is to try new complementarity functions that could be more suitable for root-finding procedures than the Alart-Curnier or De Saxcé functions.

First, we may want to look back at the scalar case, and consider the linear complementarity problem $0 \leq u \perp r \geq 0$. This scalar complementarity problem can be expressed as the normal cone inclusion $u \in -\mathcal{N}_{\mathbb{R}_+}(r)$, and thus as the zero of a complementarity function $f : (u, r) \mapsto \Pi_{\mathbb{R}_+}(r - \xi u) - r$, which looks somewhat similar to our Alart–Curnier and De Saxcé functions. However, for root-finding purposes this projection-based complementarity function is usually discarded in favor of the Fischer–Burmeister complementarity function (Burmeister 1985; Fischer 1992),

$$f_{\text{FB}} : \begin{cases} \mathbb{R}^2 & \longrightarrow \mathbb{R} \\ (u, r) & \longmapsto \sqrt{u^2 + r^2} - u - r. \end{cases}$$

The Fischer–Burmeister function is indeed "smoother"; it is only non-differentiable for $u = r = 0$ (while the projection is not-differentiable for $r = -\xi u$) and its Jacobian has been shown to be regular under reasonable conditions (Fischer 1992). Newton algorithms over the Fischer–Burmeister function have been proved to perform quite well in practice (e.g., Munson et al. 2001; Silcowitz et al. 2009).

Finding an analogous well-behaved complementarity function for SOC problems would therefore be of interest, and the Fischer–Burmeister has fortunately been extended to SOC comple-

mentarity problems by Fukushima et al. (2002). Indeed,

$$\begin{aligned} \mathscr{K}_1 \ni \boldsymbol{x} \perp \boldsymbol{y} \in \mathscr{K}_1 \iff f_{\mathrm{FB}}^{\mathscr{K}}(\boldsymbol{x},\boldsymbol{y}) = \boldsymbol{0}, \\ f_{\mathrm{FB}}^{\mathscr{K}} : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^d \\ (\boldsymbol{x},\boldsymbol{y}) \mapsto (\boldsymbol{u} \circ \boldsymbol{u} + \boldsymbol{r} \circ \boldsymbol{r})^{\frac{1}{2}} - \boldsymbol{x} - \boldsymbol{y} \end{aligned} \tag{4.1}$$

where the square-root is also defined w.r.t. the bilinear operator $\cdot \circ \cdot : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^d$,

$$\boldsymbol{x} \circ \boldsymbol{y} \mapsto \begin{pmatrix} \boldsymbol{x}^\intercal \boldsymbol{y} \\ \boldsymbol{x}_{\mathrm{N}} \boldsymbol{y}_{\mathrm{T}} + \boldsymbol{y}_{\mathrm{N}} \boldsymbol{x}_{\mathrm{T}} \end{pmatrix}.$$

The vectorial space $\mathbb{R}^d$ equipped with standard addition and the "$\circ$" product defines a Jordan (i.e., commutative, but generally not associative) algebra. The well-definition of the SOC $f_{\mathrm{FB}}^{\mathscr{K}}$ function will be asserted in a following paragraph.

Now, remember from Property 1.4 that the Coulomb law can be expressed equivalently as

$$(\boldsymbol{u},\boldsymbol{r}) \in \mathscr{C}_\mu \iff \mathscr{K}_{\frac{1}{\mu}} \ni \tilde{\boldsymbol{u}} \perp \boldsymbol{r} \in \mathscr{K}_{\frac{1}{\mu}}$$

where $\tilde{\boldsymbol{u}} = \boldsymbol{u} + \mu \|\boldsymbol{u}_{\mathrm{T}}\|$ is defined from the De Saxcé change of variable. For $\mu > 0$, we introduce two new changes of variable,

$$\hat{\boldsymbol{r}} := \begin{pmatrix} \mu \boldsymbol{r}_{\mathrm{N}} \\ \boldsymbol{r}_{\mathrm{T}} \end{pmatrix} \qquad\qquad \hat{\boldsymbol{u}} := \rho \begin{pmatrix} \tilde{\boldsymbol{u}}_{\mathrm{N}} \\ \mu \boldsymbol{u}_{\mathrm{T}} \end{pmatrix} \tag{4.2}$$

where ρ is a scalar introduced for conditioning purposes (Here as $\boldsymbol{r}$ is an impulse and $\boldsymbol{u}$ a velocity, therefore ρ should be a mass). Then,

$$\mathscr{K}_{\frac{1}{\mu}} \ni \tilde{\boldsymbol{u}} \perp \boldsymbol{r} \in \mathscr{K}_{\frac{1}{\mu}} \iff \mathscr{K}_1 \ni \hat{\boldsymbol{u}} \perp \hat{\boldsymbol{r}} \in \mathscr{K}_1$$

and thus from Equation (4.1),

$$f_{\mathrm{FB}}^{\mathscr{K}}(\hat{\boldsymbol{u}},\hat{\boldsymbol{r}}) = \boldsymbol{0} \iff (\boldsymbol{u},\boldsymbol{r}) \in \mathscr{C}_\mu. \tag{4.3}$$

This new complementarity function for Coulomb friction is only valid for $\mu > 0$, but since for $\mu = 0$ Coulomb's law degenerates to a scalar complementarity problem, in this case we can use the original, scalar Fischer–Burmeister function. The SOC $f_{\mathrm{FB}}^{\mathscr{K}}$ function can also be trivially adapted to convexified DCFP, as it is sufficient to use the original velocity $\boldsymbol{u}$ instead of the De Saxcé change of variable $\tilde{\boldsymbol{u}}$ in Equation (4.2).

Unlike what we claimed in (Daviet, Bertails-Descoubes, and Boissieux 2011), the SOC Fischer–Burmeister function has already been applied (and discarded as not efficient) by Cadoux (2009) inside a quasi-Newton algorithm. We proposed instead to use a Newton algorithm, and obtained satisfying results.

Numerical evaluation Computing the SOC $f_{\mathrm{FB}}^{\mathscr{K}}$ function involves taking the square-root of a vector of $\mathbb{R}^d$ w.r.t. the bilinear operator $\cdot \circ \cdot$; one may wonder about the well-definedness of this operation, and its ease of computation.

The association of the vector space $\mathbb{R}^d$ with the bilinear operator $\circ$ forms a $\mathbb{R}$-algebra, tightly associated to the SOC $\mathscr{K}_1$ through two notable properties:

- $\forall \boldsymbol{x} \in \mathscr{K}_1$, there exists a unique $\boldsymbol{z} \in \mathscr{K}_1$ s.t. $\boldsymbol{z} \circ \boldsymbol{z} = \boldsymbol{x}$; we note $\boldsymbol{z} = \boldsymbol{x}^{\frac{1}{2}}$;
- $\forall \boldsymbol{x} \in \mathbb{R}^d$, $\boldsymbol{x} \circ \boldsymbol{x} \in \mathscr{K}_1$.

As $\mathscr{K}_1$ is a convex cone, we deduce that $\forall \boldsymbol{x},\boldsymbol{y} \in \mathbb{R}^d \times \mathbb{R}^d$, $\boldsymbol{x} \circ \boldsymbol{x} + \boldsymbol{y} \circ \boldsymbol{y} \in \mathscr{K}_1$ and thus the SOC Fischer–Burmeister function is well-defined.

Moreover, any vector $\boldsymbol{x} \in \mathbb{R}^d$ can be decomposed as $\boldsymbol{x} = \lambda_1 \boldsymbol{v}_1 + \lambda_2 \boldsymbol{v}_2$, where

$$\lambda_i = \boldsymbol{x}_\mathrm{N} + (-1)^i \|\boldsymbol{x}_\mathrm{T}\| \tag{4.4}$$

$$\boldsymbol{v}_i = \frac{1}{2} \begin{cases} \left(1; (-1)^i \dfrac{\boldsymbol{x}_\mathrm{T}}{\|\boldsymbol{x}_\mathrm{T}\|}\right) & \text{if } \boldsymbol{x}_\mathrm{T} \neq \boldsymbol{0} \\ \left(1; (-1)^i \boldsymbol{e}\right) & \text{if } \boldsymbol{x}_\mathrm{T} = \boldsymbol{0} \end{cases} \tag{4.5}$$

where $\boldsymbol{e}$ is any unit vector in $\mathbb{R}^{d-1}$. λ_1 and λ_2 are called the *eigenvectors* of $\boldsymbol{x}$, and $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ are the associated *eigenvalues*. If $\boldsymbol{x} \in \mathscr{K}_1$, λ_1 and λ_2 will both be positive; we can thus define $\boldsymbol{z} := \sqrt{\lambda_1}\boldsymbol{v}_1 + \sqrt{\lambda_2}\boldsymbol{v}_2$, and check that $\boldsymbol{z} \circ \boldsymbol{z} = \boldsymbol{x}$. This yields an easy access to the square-root of any element of $\mathscr{K}_1$, and thus to the SOC Fischer–Burmeister function $f_\mathrm{FB}^{\mathscr{K}}$.

Newton algorithm The $f_\mathrm{FB}^{\mathscr{K}}$ function is Lipschitz-continuous, as is the gradient of its squared norm, $(\partial f_\mathrm{FB}^{\mathscr{K}})^\intercal f_\mathrm{FB}^{\mathscr{K}}$, suggesting good conditions for a Newton minimization algorithm on $\Psi_\mathrm{FB} := \frac{1}{2}\|f_\mathrm{FB}^{\mathscr{K}}(\hat{\boldsymbol{u}}(\boldsymbol{r}), \hat{\boldsymbol{r}})\|^2$. Moreover, $f_\mathrm{FB}^{\mathscr{K}}$ is *strongly semismooth* (D. Sun and J. Sun 2005), justifying performing directly a nonsmooth root-finding Newton algorithm (Qi and J. Sun 1993) on $f_\mathrm{FB}^{\mathscr{K}}(\hat{\boldsymbol{u}}(\boldsymbol{r}), \hat{\boldsymbol{r}}) = \boldsymbol{0}$, as this avoids computing the second derivative of $f_\mathrm{FB}^{\mathscr{K}}$.

The Newton algorithm implemented in the `bogus` library is a simplification of the one that we suggested originally (Daviet, Bertails-Descoubes, and Boissieux 2011, Appendix A.2), and is given in Appendix B.1.2; the derivatives of $f_\mathrm{FB}^{\mathscr{K}}$ are provided in Appendix B.1.1.

We observed that on average, using the SOC Fischer–Burmeister complementarity function made the Gauss–Seidel algorithm perform slightly better than using the Alart–Curnier formulation (see Table 4.3). However, this was still not sufficient to achieve convergence for every local problem. Damping the Newton algorithms (by performing a line-search) improves their robustness, at the cost of being much slower in the general case, and still failing sometimes. More precisely, on our hair problems the undamped Fischer–Burmeister Newton (Algorithm B.1) was much more robust that the undamped Alart–Curnier Newton, and slightly less robust but much faster that the undamped Alart–Curnier Newton with line-search (Table 4.3 below).

Building failsafes with different combinations of root-finding algorithms on complementarity functions did not yield significantly better results. Instead, we have chosen to conceive a failsafe using a radically different approach, solving the one-contact problem analytically.

4.1.2 Analytical solver

The idea of solving the one-contact friction problems analytically is not new. In 2D, it was mentioned by Klarbring (1990), and used inside a Gauss–Seidel algorithm by Jean (1999). In 3D however, this method appears to be surprisingly less popular, and we could not find any evidence of previous usage in numerical codes.

Analytical solvers attempt to solve problem (3.11) by trying to find a solution in each case of the disjunctive formulation (1.1). Here we will assume that W is symmetric positive-definite; if not, we can simply use the proximal variant (Section 3.4.2) of the Gauss–Seidel algorithm to alleviate this difficulty.

The enumerative algorithm proceeds as follow:

1. We first check the easiest case, take-off. If $\boldsymbol{b}_\mathrm{N} \geq 0$, then it is sufficient to choose $\boldsymbol{r} = 0$ and $\boldsymbol{u} = \boldsymbol{b}$ for $\boldsymbol{u}$ and $\boldsymbol{r}$ to satisfy Coulomb's law.

2. Then, we check the sticking case, i.e., $\boldsymbol{u} = \boldsymbol{0}$. This means finding $\boldsymbol{r} \in \mathscr{K}_\mu$ such that $W\boldsymbol{r} + \boldsymbol{b} = \boldsymbol{0}$. As W is positive-definite, this amounts to checking that $W^{-1}\boldsymbol{b} \in -\mathscr{K}_\mu$.

3. Finally, the sliding case is the hard one, as we will see in the next paragraph.

Sliding case We try to find a solution to the sliding case of the one-contact problem,

$$\begin{cases} \begin{pmatrix} 0 \\ -\alpha \boldsymbol{r}_{\mathrm{T}} \end{pmatrix} = W\boldsymbol{r} + \boldsymbol{b} \quad \alpha \in \mathbb{R}_+ \\ \|\boldsymbol{r}_{\mathrm{T}}\| = \mu \boldsymbol{r}_{\mathrm{N}}. \end{cases} \tag{4.6}$$

In $2D$, the second row of (4.6) imposes $\boldsymbol{r}_{\mathrm{T}} = \pm\mu\boldsymbol{r}_{\mathrm{N}}$. Plugging this into the first equation, we get $(W_{\mathrm{NN}} \pm \mu W_{\mathrm{TN}})\boldsymbol{r}_{\mathrm{N}} = -\boldsymbol{b}_{\mathrm{N}}$, and it suffices to check if any of the two possibilities yields a positive α.

For the 3D case, we remark in (Bonnefon and Daviet 2011) that the null normal velocity condition $\boldsymbol{u}_{\mathrm{N}} = 0$ together with the cone boundary condition $\|\boldsymbol{r}_{\mathrm{T}}\| = \mu\boldsymbol{r}_{\mathrm{N}}$ state that $\boldsymbol{r}$ should lie in the intersection of a plane and a hollow cone, which is an ellipse. This ellipse can be parameterized by a polar angle θ, and plugging-in the $\boldsymbol{u}_{\mathrm{T}} = -\alpha\boldsymbol{r}_{\mathrm{T}}$ equation allows to reduce the potential values of θ to the roots of a quartic polynomial, which can be enumeratively checked. Alternatively, we propose also in (Bonnefon and Daviet 2011) to look for the solutions to the sliding case in the roots of another degree-four polynomial on α. This second polynomial may have a worse conditioning, but its coefficients are simpler to compute. Indeed, let us decompose W as

$$W := \begin{pmatrix} W_{\mathrm{NN}} & W_{\mathrm{TN}}^{\intercal} \\ W_{\mathrm{TN}} & W_{\mathrm{TT}} \end{pmatrix}$$

We define

$$\begin{array}{llllll} T := & W_{\mathrm{TT}} - \frac{1}{W_{\mathrm{NN}}} W_{\mathrm{TN}} W_{\mathrm{TN}}^{\top} & \mathrm{A} := & \mathrm{Tr}\, T - W_{\mathrm{TN}}^{\intercal}\boldsymbol{b}_{\mathbf{T}} & \mathrm{C} := & \det T - W_{\mathrm{TN}}^{\intercal}\mathrm{B} \\ b_{\mathbf{T}} := & \frac{\boldsymbol{b}_{\mathbf{T}}}{\boldsymbol{b}_{\mathrm{N}}} - \frac{W_{\mathrm{TN}}}{W_{\mathrm{NN}}} & \mathbf{B} := & (\mathrm{adj}\, T)\, \boldsymbol{b}_{\mathbf{T}} & \mathrm{D} := & \left(\frac{W_{\mathrm{NN}}}{\mu}\right)^2 \end{array}$$

where $\mathrm{adj}\, T$ designate the adjugate of T, that is, the transpose of its cofactor matrix. The solutions of the sliding problem (4.6) must then satisfy $P(\alpha) = 0$, with

$$P(\alpha) = \alpha^4 + 2\mathrm{A}\alpha^3 + \left(2\mathrm{C} + \mathrm{A}^2 - \mathrm{D}\boldsymbol{b}_{\mathbf{T}}^2\right)\alpha^2 + 2\left(\mathrm{CA} - \mathrm{D}\boldsymbol{b}_{\mathbf{T}}^{\intercal}\mathbf{B}\right)\alpha + \mathrm{C}^2 - \mathrm{D}\mathbf{B}^2. \tag{4.7}$$

Convexified DCFP The same approach can be used for the local problems of the convexified DCFP. The enumerative algorithm becomes:

1. ("take-off" case) If $\boldsymbol{b} \in \mathscr{K}_{\frac{1}{\mu}}$, then take $\boldsymbol{r} = \mathbf{0}$.
2. ("sticking" case) If $W^{-1}\boldsymbol{b} \in -\mathscr{K}_{\mu}$, take $\boldsymbol{r} = -W^{-1}\boldsymbol{b}$.
3. ("sliding" case) Check for solutions satisfying Equation (4.8),

$$\mathrm{Bd}\,\mathscr{K}_{\frac{1}{\mu}} \ni W\boldsymbol{r} + \boldsymbol{b} \perp \boldsymbol{r} \in \mathrm{Bd}\,\mathscr{K}_{\mu}. \tag{4.8}$$

Once again, we will be able to obtain 3D solutions by analyzing the roots of a quartic polynomial.

First, suppose that μ is zero, then $\boldsymbol{r} \in \mathrm{Bd}\,\mathscr{K}_0 \implies \boldsymbol{r}_{\mathbf{T}} = \mathbf{0}$ and therefore $\boldsymbol{r} = \left[-\frac{\boldsymbol{b}_{\mathrm{N}}}{W_{\mathrm{NN}}}; \mathbf{0}\right]$ is a solution since $\boldsymbol{b}_{\mathrm{N}} < 0$ (otherwise we would have $\boldsymbol{b} \in \mathscr{K}_{\infty}$ and therefore be in the "take-off" case).

Now we assume that $\mu > 0$. We are looking for solutions where $\boldsymbol{r}_{\mathrm{N}}$ and $\boldsymbol{u}_{\mathrm{N}}$ are non zero, as $\boldsymbol{u} = \mathbf{0}$ and $\boldsymbol{r} = \mathbf{0}$ already correspond to "take-off" and "sliding" cases. Such solutions must satisfy

$$\begin{cases} \boldsymbol{u} = W\boldsymbol{r} + \boldsymbol{b} \\ \|\boldsymbol{r}_{\mathrm{T}}\| = \mu\boldsymbol{r}_{\mathrm{N}} \\ \|\boldsymbol{u}_{\mathrm{T}}\| = \frac{1}{\mu}\boldsymbol{u}_{\mathrm{N}} \\ \exists \alpha > 0 \quad s.t. \quad \boldsymbol{u}_{\mathbf{T}} = -\alpha\boldsymbol{r}_{\mathbf{T}}. \end{cases} \tag{4.9}$$

In the 2D case, this means $\boldsymbol{r}_{\mathrm{T}} = \pm\mu\boldsymbol{r}_{\mathrm{N}}$. We can successively look for a solution in both cases; with $s := \mathrm{sign}(\boldsymbol{r}_{\mathrm{T}})$, this amounts to solving

$$W_{\mathrm{NN}} + 2\mu s W_{\mathrm{TN}} + \mu^2 W_{\mathrm{TT}}\boldsymbol{r}_{\mathrm{N}} = -(\boldsymbol{b}_{\mathrm{N}} + \mu s \boldsymbol{b}_{\mathrm{T}}) \tag{4.10}$$

and checking that the resulting $\boldsymbol{r}_{\mathrm{N}}$ and $\boldsymbol{u}_{\mathrm{N}}$ are both positive.

Now let us consider the 3D case, and write $(\boldsymbol{e}_{\mathrm{N}}, \boldsymbol{e}_{\mathrm{T}_1}, \boldsymbol{e}_{\mathrm{T}_2})$ the canonical basis of $\mathbb{R}^3$. Writing Equation (4.9) with cylindrical coordinates yields

$$\left\{\begin{array}{rcl} \boldsymbol{u} & = & W\boldsymbol{r} + \boldsymbol{b} \\ \boldsymbol{r} & = & r\left(\mathbf{e}_N + \mu\left(\cos\theta\boldsymbol{e}_{\mathrm{T}_1} + \sin\theta\boldsymbol{e}_{\mathrm{T}_2}\right)\right) \\ \boldsymbol{u} & = & u\left(\mu\mathbf{e}_N - \left(\cos\theta\boldsymbol{e}_{\mathrm{T}_1} + \sin\theta\boldsymbol{e}_{\mathrm{T}_2}\right)\right) \\ u > 0 & , & r > 0. \end{array}\right. \tag{4.11}$$

The first line of (4.11) reads

$$u\begin{pmatrix}\mu\\ \cos\theta\\ \sin\theta\end{pmatrix} - r\begin{pmatrix}\mu\cos\theta W_{\mathrm{NT}_1} + \mu\sin\theta W_{\mathrm{NT}_2} + W_{\mathrm{NN}}\\ \mu\cos\theta W_{\mathrm{T}_1\mathrm{T}_1} + \mu\sin\theta W_{\mathrm{T}_1\mathrm{T}_2} + W_{\mathrm{NT}_1}\\ \mu\cos\theta W_{\mathrm{T}_1\mathrm{T}_2} + \mu\sin\theta W_{\mathrm{T}_2\mathrm{T}_2} + W_{\mathrm{NT}_2}\end{pmatrix} = \boldsymbol{b}. \tag{4.12}$$

We can eliminate u by plugging the first row into the second and third ones,

$$r\begin{pmatrix}-\cos^2\theta W_{\mathrm{NT}_1} + \cos\theta\sin\theta W_{\mathrm{NT}_2} + \left(\cos\theta W_{\mathrm{T}_1\mathrm{T}_1} + \sin\theta W_{\mathrm{T}_1\mathrm{T}_2}\right)\mu + \frac{\cos\theta W_{\mathrm{NN}}}{\mu} + W_{\mathrm{NT}_1}\\ -\cos\theta\sin\theta W_{\mathrm{NT}_1} + \sin^2\theta W_{\mathrm{NT}_2} + \left(\cos\theta W_{\mathrm{T}_1\mathrm{T}_2} + \sin\theta W_{\mathrm{T}_2\mathrm{T}_2}\right)\mu + \frac{\sin\theta W_{\mathrm{NN}}}{\mu} + W_{\mathrm{NT}_2}\end{pmatrix}$$
$$= \boldsymbol{b}_{\mathrm{T}} + \frac{1}{\mu}\boldsymbol{b}_{\mathrm{N}}, \begin{pmatrix}\cos\theta\\ \sin\theta\end{pmatrix}$$

then eliminate r by subtracting these two equations scaled by respectively $\boldsymbol{b}_{\mathrm{T}_2} + \frac{b_{\mathrm{N}}\sin\theta}{\mu}$ and $\boldsymbol{b}_{\mathrm{T}_1} + \frac{b_{\mathrm{N}}\cos\theta}{\mu}$, to obtain the necessary condition (4.13),

$$\begin{array}{rl} 0 = & \mu\left(\boldsymbol{b}_{\mathrm{T}_1}W_{\mathrm{NT}_1} - \boldsymbol{b}_{\mathrm{T}_2}W_{\mathrm{NT}_2} - \boldsymbol{b}_{\mathrm{N}}W_{\mathrm{T}_1\mathrm{T}_1} + \boldsymbol{b}_{\mathrm{N}}W_{\mathrm{T}_2\mathrm{T}_2}\right)\cos\theta\sin\theta \\ & -\left(\left(\boldsymbol{b}_{\mathrm{T}_2}W_{\mathrm{T}_1\mathrm{T}_2} - \boldsymbol{b}_{\mathrm{T}_1}W_{\mathrm{T}_2\mathrm{T}_2}\right)\mu^2 + \boldsymbol{b}_{\mathrm{T}_1}W_{\mathrm{NN}} - \boldsymbol{b}_{\mathrm{N}}W_{\mathrm{NT}_1}\right)\sin\theta \\ & -\left(\boldsymbol{b}_{\mathrm{T}_2}W_{\mathrm{NT}_1} - \boldsymbol{b}_{\mathrm{N}}W_{\mathrm{T}_1\mathrm{T}_2}\right)\cos^2\theta\mu + \left(\boldsymbol{b}_{\mathrm{T}_1}W_{\mathrm{NT}_2} - \boldsymbol{b}_{\mathrm{N}}W_{\mathrm{T}_1\mathrm{T}_2}\right)\sin^2\theta\mu \\ & -\left(\left(\boldsymbol{b}_{\mathrm{T}_2}W_{\mathrm{T}_1\mathrm{T}_1} - \boldsymbol{b}_{\mathrm{T}_1}W_{\mathrm{T}_1\mathrm{T}_2}\right)\mu^2 + \boldsymbol{b}_{\mathrm{T}_2}W_{\mathrm{NN}} - \boldsymbol{b}_{\mathrm{N}}W_{\mathrm{NT}_2}\right)\cos\theta \\ & -\left(\boldsymbol{b}_{\mathrm{T}_2}W_{\mathrm{NT}_1} - \boldsymbol{b}_{\mathrm{T}_1}W_{\mathrm{NT}_2}\right)\mu. \end{array} \tag{4.13}$$

Using the half-angle change of variable $t = \tan\frac{\theta}{2}$, we obtain a degree-4 polynomial in t whose roots yield a superset of the solutions to Equation (4.9). We then just need to look for any of the at most 4 roots that satisfies the original equation.

For the sake of completeness, we include below the expression of the coefficients of this polynomial, obtained thanks to the Sage[1] computer algebra system. Let

$$\begin{array}{rcl} A & := & \left(\boldsymbol{b}_{\mathrm{T}_1}W_{\mathrm{T}_1,\mathrm{T}_2} - \boldsymbol{b}_{\mathrm{T}_2}W_{\mathrm{T}_1,\mathrm{T}_1}\right)\mu^2 + \boldsymbol{b}_{\mathrm{N}}W_{\mathrm{N},\mathrm{T}_2} - \boldsymbol{b}_{\mathrm{T}_2}W_{\mathrm{N},\mathrm{N}} \\ B & := & \left(\boldsymbol{b}_{\mathrm{N}}W_{\mathrm{T}_1,\mathrm{T}_2} + \boldsymbol{b}_{\mathrm{T}_1}W_{\mathrm{N},\mathrm{T}_2} - \mathrm{T}_2\boldsymbol{b}_{\mathrm{T}_2}W_{\mathrm{N},\mathrm{T}_1}\right)\mu \\ C & := & 2\left(\left(\boldsymbol{b}_{\mathrm{T}_1}W_{\mathrm{T}_2,\mathrm{T}_2} - \boldsymbol{b}_{\mathrm{T}_2}W_{\mathrm{T}_1,\mathrm{T}_2}\right)\mu^2 - \boldsymbol{b}_{\mathrm{N}}W_{\mathrm{N},\mathrm{T}_1} + \boldsymbol{b}_{\mathrm{T}_1}W_{\mathrm{N},\mathrm{N}}\right) \\ D & := & 2\left(\boldsymbol{b}_{\mathrm{N}}\left(W_{\mathrm{T}_1,\mathrm{T}_1} - W_{\mathrm{T}_2,\mathrm{T}_2}\right) - \boldsymbol{b}_{\mathrm{T}_1}W_{\mathrm{N},\mathrm{T}_1} + \boldsymbol{b}_{\mathrm{T}_2}W_{\mathrm{N},\mathrm{T}_2}\right)\mu \\ E & := & -6\left(\boldsymbol{b}_{\mathrm{N}}W_{\mathrm{T}_1,\mathrm{T}_2} - \boldsymbol{b}_{\mathrm{T}_1}W_{\mathrm{N},\mathrm{T}_2}\right)\mu, \end{array}$$

then Equation (4.13) becomes equivalent to $P(t) = 0$ with

$$P(t) := (B - A)t^4 + (C + D)t^3 + Et^2 + (C - D)t + A + B. \tag{4.14}$$

[1] https://www.sagemath.org

Quartic solvers We need an efficient and precise way of finding the roots of the quartic polynomials from the sliding cases, (4.7) and (4.14). There exists a few analytical algorithms for finding the roots of a quartic polynomial, the first one being attributed to Lodovico Ferrari in 1540. Such algorithms are complex, as they involve accounting for a lot of potential cases; luckily, open-source implementations have been written, for instance in the GNU Scientific Library (GSL)[2]. If a perfectly analytical solution is not required, one may also use a general purpose polynomial root-finding algorithm, such as the famous Netlib routine `rpoly`[3], or compute the eigenvalues of the companion matrix. Despite performing more floating point operations, these latter solutions are often faster in practice, thanks to a lower number of conditionals.

Hybrid local solver Using our new analytical solver inside the Gauss–Seidel algorithm proved to be quite robust, but once again did not fully eliminate the simulation failures (see Table 4.3). The main drawback of the enumerative solver is that, in cases where no solution is found, no approximate solution is computed either, as would be the case for iterative solvers. Even if a problem is very close to have a solution (say, for instance that $\boldsymbol{b}_{\mathrm{N}}$ is slightly below 0 due to numerical errors and no solution is found in either the sliding or sticking case), the analytical solver will simply state that no solution exists. In contrast, in this same scenario, a Newton solver would simply return a zero force vector and yield a very small residual. One solution could be to add tolerances when checking for each case in the enumerative algorithm; but that would just amount to slightly postponing the difficulty. We found that one more satisfying solution was to combine the two different approaches, enumerative and iterative, thus benefitting from the best of both strategies.

In (Daviet, Bertails-Descoubes, and Boissieux 2011), we computed the quartic's roots analytically using the GSL's algorithm. As this was quite slow, we used the enumerative solver only as a failsafe for the Fischer–Burmeister Newton algorithm. In `bogus`, a slightly different approach is used: the enumerative solver is first used, computing the quartic's roots as the eigenvalues of the polynomial's companion matrix. For this, we use Eigen's built-in solver (Guennebaud, Jacob, et al. 2010), which proved to be quite fast. Then, the Newton solver is unconditionally applied. The computed eigenvalues might not be extremely precise, but will still be close enough to the solution so that the Newton algorithm will only need a very low number of iterations (usually 0 or 1) to bring the error below the required tolerance. If no solution is found by the enumerative algorithm, then the Newton algorithm is simply started from the previous iteration solution.

Both approaches work well in practice, proving themselves both more robust and more efficient than any non-hybrid solution. From a practical point of view, however, the second strategy may be more convenient to implement.

4.1.3 Full algorithm

Now that we have a robust and efficient one-contact solver, let us focus on the global Gauss–Seidel algorithm. At each iteration, the canonical algorithm solves the local problem for each contact. However, we observe that the speed of convergence of the local forces is highly inhomogeneous; some contacts will reach their final value after a few iterations, with others will require dozens of them. For this reason, we would rather focus our computing power on the contacts that are slow to converge.

Sleeping heuristics Radjai et al. (1998) observe that the intensity of the contact forces inside a granular media follows a bimodal distribution. They describe the emergence of *force chains*, that is, a percolating load-bearing structure consisting of sticking contacts with high forces, and local dissipative regions where the forces are much smaller and where sliding occurs. Using this insight, we want to focus on getting the force chains right; the dissipative regions, being local and having less influence on the force network, should require less iterations to converge. We

[2] https://www.gnu.org/software/gsl/
[3] http://netlib.sandia.gov/port/prop.upd/rpoly.f

Figure 4.1: *Comparison of the hair collective behavior between (top) real hair motion sequences and (bottom) our corresponding simulations, based on large assemblies of (up to 2,000) individual fibers with contacts and Coulomb friction.*

propose a *contact "sleeping"* heuristic, which temporarily stops solving contacts at which the force remains roughly constant or is small. The contacts are then reactivated after a few iterations, 10 in our implementation. This simple heuristic leads to significant performance improvements (Figure 4.8), and we can show that it does not affect the theoretical convergence results of the Gauss–Seidel applied to the minimization of a convex function (Appendix B.3).

Stopping criterion The standard stopping criterion for Gauss–Seidel algorithms is to look at the difference in forces (or velocities) between one step and the next, and exit when this difference is small enough. Yet, we did not find this criterion to be always representative of the quality of the current iterate, as small steps could simply indicate failed local solves. We choose instead to make use of our Fischer–Burmeister complementarity function, and to evaluate the error as the maximum of the f_{FB} norm over all contact points. However, this requires computing the current relative velocity as $\boldsymbol{u} = W\boldsymbol{r} + \boldsymbol{b}$. This costs one matrix–vector operation, which is as much as the cost of computing $\bar{\boldsymbol{b}}_i$ for the whole set of contacts. Using our error measurement at each Gauss–Seidel iteration would therefore mean almost doubling the total cost of our algorithm, which is unacceptable. Instead, we evaluate the residual only every N iterations ($N = 25$ in our case), so that the overhead becomes negligible.

Final algorithm Our final algorithm for solving the DCFP or convexified DCFP (more specifically the matrix-free version, which can be easily adapted for an Delassus operator W) is laid out in Algorithm 4.1; the exact implementation can be found is the `bogus` library.

4.2 Application to hair dynamics

Modeling hair dynamics is challenging: human hair is composed of about 150,000 individual fibers that tightly interact together, leading to a complex collective behavior. Due to the rough surface of the hair fibers, covered with microscopic scales, dry friction is substantial at the contact points, and consequently greatly influences the hair dynamics at the *macroscopic* level. As illustrated in Figure 4.2, we have identified in (Daviet, Bertails-Descoubes, and Boissieux 2011) three major hair visual features that directly emerge from those nonsmooth frictional contacts:

1. Stick-slip instabilities during motion;
2. Spontaneous splitting of hair into multiple untidy wisps and "flyers" during strong motion *vs.* its spontaneous grouping into a few globally coherent locks during gentle motion;
3. Appearance of complex hair patterns that can remain perfectly still at the end of the motion.

We claim that accounting for these phenomena is essential for producing realistic and compelling hair animations.

Algorithm 4.1: Matrix-free Gauss–Seidel algorithm with sleeping heuristics

input : Initial guess $\boldsymbol{r}$
input : Matrices M, H and P
input : Vectors $\boldsymbol{b}$ and $\boldsymbol{c}$
input : $\boldsymbol{\beta}$ array of proximal coefficients
Result: $\boldsymbol{r}$ and $\boldsymbol{\lambda}$ approximate solutions to the DCFP

$\boldsymbol{l} \leftarrow (HM^{-1}C^{\intercal})P^{-1}\left(\boldsymbol{c} - (HM^{-1}C^{\intercal})^{\intercal}\boldsymbol{r}\right)$; `// Linear constraints`
for *i=1* **to** *maxIters* **do**
  SkipTab [i] $\leftarrow 0$; `// Reset sleeping contacts counter`
  $W_{ii} \leftarrow \sum_j^m H_{ij}M_{jj}H_{ij}^{\intercal} + \boldsymbol{\beta}_i\mathbb{I}$; `// Precompute local matrices`
end
$\boldsymbol{z} \leftarrow M^{-1}H^{\intercal}\boldsymbol{r}$;
for $k \leftarrow 1$ **to** *maxIters* **do** `// GS iteration`
  for $i \leftarrow 1$ **to** n **do** `// Loop over contacts`
    if SkipTab *[i] > 0* **then** `// Sleeping contact`
      SkipTab $[i] \leftarrow$ SkipTab $[i] - 1$;
      continue;
    end
    $\bar{\boldsymbol{b}}_i \leftarrow \boldsymbol{b}_i + \boldsymbol{l}_i + H.\text{row}(i)\ \boldsymbol{z}$ - $W_{ii}\boldsymbol{r}_i$;
    $\boldsymbol{r}_i^{\text{prev}} \leftarrow \boldsymbol{r}_i$;
    solutionFound $\leftarrow$ `solveLocalProblem(` W_{ii}, $\bar{\boldsymbol{b}}_i$, $\boldsymbol{r}_i$) ;
    if not *solutionFound* **then** $\boldsymbol{r} \leftarrow \frac{1}{2}\left(\boldsymbol{r}_i + \boldsymbol{r}_i^{\text{prev}}\right)$;
    $\boldsymbol{z} \leftarrow \boldsymbol{z} + (M^{-1}H).\text{col}(i)\ (\boldsymbol{r}_i - \boldsymbol{r}_i^{\text{prev}})$;
    if $\|\boldsymbol{r}_i\|^2 < \epsilon_{small}$ **or** $\|\boldsymbol{r}_i - \boldsymbol{r}_i^{prev}\|^2 < \epsilon_{converged}$ **then**
      SkipTab [i] $\leftarrow$ *nSkip* ; `// Put contact to sleep`
    end
  end
  $\boldsymbol{l} \leftarrow (HM^{-1}C^{\intercal})P^{-1}\left(\boldsymbol{c} - (HM^{-1}C^{\intercal})^{\intercal}\boldsymbol{r}\right)$; `// Linear constraints`
  if k **mod** *nEvalEvery* $= 0$ **then**
    `// Evaluate residual`
    $\boldsymbol{z} \leftarrow M^{-1}H^{\intercal}\boldsymbol{r}$;
    $\boldsymbol{u} \leftarrow H\boldsymbol{z} + \boldsymbol{b} + \boldsymbol{l}$;
    if $\max_i \|f_{FB}(\boldsymbol{u}_i, \boldsymbol{r}_i)\|^2 < tol$ **then break**;
  end
end

Figure 4.2: *Three important features of the real hair collective behavior, emerging from nonsmooth friction: (1) Stick-slip instabilities, especially visible here between the hair and the right shoulder; (2) Spontaneous splitting of hair into thin wisps and "flyers" during strong motion (left)* vs. *spontaneous grouping of hair into a few coherent locks during gentle motion (middle and right); (3) Complex hair patterns that may remain stable at rest. Courtesy of Sylvain Paris and Tilke Judd, MIT 2007.*

4.2.1 Hair simulation in Computer Graphics

Due to performance limitations, the first works that attempted to simulate hair dynamics mostly neglected hair self-interactions (Anjyo et al. 1992; Rosenblum et al. 1991), or processed them at a coarse level between a small number of predefined interacting wisps using penalty forces (Bertails et al. 2006; Choe et al. 2005; Plante et al. 2001). Alternatively, Hadap and Magnenat-Thalmann (2001) proposed a macroscopic model of the hair medium based on a fluid solver, observing that some fluid properties, such as incompressibility, could be representative of the hair collective behavior. Though interesting, their approach fails to capture the discontinuities emerging from large hair motions. Thanks to the design of realistic, robust and fast primitives for thin elastic rods (Bergou, Audoly, et al. 2010; Bergou, Wardetzky, et al. 2008; Bertails et al. 2006; Hadap 2006; Pai 2002; Selle et al. 2008; Spillmann and Teschner 2007), some approaches have been developed at the fiber lever in order to gain realism in hair simulations.

Selle et al. (2008) designed an efficient mass-spring model for an individual fiber, allowing them to simulate up to 10,000 fibers in a reasonable computational time (from a few minutes up to one hour per frame) on a quad-core architecture. Unfortunately, many self-contacts were ignored, causing the fibers to penetrate each other, and thus failing to preserve the hair volume. To resolve these issues while still retaining some discontinuous details in the simulations, McAdams et al. (2009) proposed a hybrid Eulerian/Lagrangian hair model combining a fluid model together with the explicit treatment of fiber self-contacts. With this approach, the hair volume is properly preserved while detailed interactions at the fiber level yield nice visual effects. However, nonsmooth effects due to dry friction, which play a major role in hair dynamics, are not captured.

In (Daviet, Bertails-Descoubes, and Boissieux 2011), we used the Super-Helix (Bertails et al. 2006) fiber model, which is a reduced-coordinate discretization of a Kirchhoff rod whose sole degrees of freedom are piecewise-constant curvatures and twists. This model possesses two main

Name	N_{rods}	$\mu_{\text{hair}}/\mu_{\text{body}}$	$\overline{N_{\text{contacts}}}$	$\max(N_{\text{contacts}})$
Free/A	1920	0.3 / 0.5	24659	46915
Free/B	1920	0.2 / 0.3	28153	36287
Pony	334	0.3 / 0.5	9850	16613
Curly	1920	0.3 / 0.5	21425	33578

Overlines indicate averaged quantities.

Table 4.1: *Physical properties of our hair simulations.*

Name	$\%_{>\text{tol}}$ [1]	$\max(err)$	$\overline{\text{iters}}$	$\overline{T_{\text{GS}}}$ / $\overline{T_{\text{solv}}}$ [2] (s)	$\overline{T_{dt}}$ [3] (s)	$\overline{T_{\text{frame}}}$ (min)	T_{tot} (hours)
Free/A	0.056	0.004	136	2.60 / 4.06	7.15	2.09	25
Free/B	0.13	0.037	160	5.3 / 7.0	11.1	3.02	36
Pony	0.42	0.003	301	4.43 / 5.25	6.2	2.47	30
Curly	0.013	0.003	118	2.26 / 3.39	7.54	4.15	49

Overlines indicate averaged quantities.
[1] $\%_{>\text{tol}}$: Percentage of one-step problems that did not reach (global) tolerance
[2] T_{GS} / T_{solv}: Time for the Gauss–Seidel loop (Algorithm 4.1) alone / Total time for the contact solver (including the assembly of W)
[3] T_{dt}: Total time for one simulation timestep (dt ranges from 1 to 4ms)

Table 4.2: *Performance results for our hair simulations.*

advantages: the inextensibility of the fibers is intrinsically enforced by the choice of coordinates without requiring a supplemental holonomic constraint nor stiff springs, and the bending forces are linear in the degrees of freedom, allowing for easy implicit integration. Drawbacks include having to use a limited number of degrees of freedom, as the stiffness matrices of the fibers are dense and expensive to compute, and vanishing inertia terms in straight configurations, leading to potentially degenerate systems. We treated contacts using the Moreau-Jean timestepping scheme (Section 2.2.2) and our Gauss–Seidel solver (Section 4.1). This allowed us to simulate a few thousand individual fibers at a few minutes per frame, capturing the variety of effects induced by static friction.

More recently, Iben et al. (2013) used penalties for hair–hair contacts as in (Selle et al. 2008), but did not limit the maximum number of contacts and used a contact-pruning algorithm specifically designed to maintain the volume of stylized hair. Other authors have also leveraged our hybrid solver. Aubry and Xian (2015) proposed to extend the Discrete Elastic Rods (DER; Bergou, Audoly, et al. 2010; Bergou, Wardetzky, et al. 2008) model with root constraints to perform implicit simulation of flexible trees, leveraging our hybrid Gauss–Seidel algorithm to deal with frictional contacts. Kaufman, Tamstorf, et al. (2014) devised a Newton algorithm to adaptively handle the nonlinearities of the DER model. Thanks to a matrix-free variant of our Gauss–Seidel algorithm, they were able to simulate scenes with tens of thousands of fibers and up to a million contact points per timestep. In a similar manner, Gornowicz and Borac (2015) used a Newton algorithm over a variation of the Discrete Elastic Rod model with a modified elastic energy, but with a linear approximation of the friction cone.

4.2.2 Full-scale simulations

To evaluate the effectiveness of our contact solver (Algorithm 4.1) on hair simulations, we ran three kinds of experiments, summarized in Table 4.1. All are using the Super-Helix model, with 16 degrees of freedom per rod. The character was animated using 3ds Max (Autodesk 2009) by reproducing a real video-captured motion that serves as a reference (see Figure 4.1 and the accompanying video[4]). The hair simulation entitled “Free” models a full, unconstrained haircut, and consists of about 2000 simulated rods. It is divided into two parts, “A” and “B”, featuring a

[4] http://bipop.inrialpes.fr/~bertails/Papiers/Videos/hairContactSiggraphAsia2011.mp4

head rotation motion and a head leaning motion, respectively. The third hair simulation, entitled "Pony", contains only slightly over 300 rods, but those are tightly packed into a ponytail. All of these experiments include smooth as well as rough head motions. Finally, a last motion, "Curly', illustrates the fact that our method can also easily handle curly hair (see Figure 4.5). It is based on the "Free/A" head motion.

Figure 4.3: *Simulation of a fast head movement without (top) and with (bottom) Coulomb friction. In the latter case, hair remains much more coherent and the results are visually more appealing.*

Visual results and comparisons to real hair motions are presented in Figures 4.1, 4.3, 4.4 and 4.5. Final rendering was performed using 3ds Max. Our method allows us to preserve the hair volume and to capture subtle phenomena such as stick-slip instabilities or the spontaneous appearance of transient coherent movements in hair. We also observe that a lot of energy is dissipated by Coulomb friction; capturing it accurately is essential to achieve realism. Figure 4.3 shows the effect of decreasing the hair/hair friction coefficient on the collective hair behavior. Without friction, hair looks artificially clean and light. In contrast, in the presence of friction, motion looks coherent and properly damped, while simultaneously featuring complex details at the fiber level.

Measure of performance and robustness All our simulations were run on a desktop machine featuring a Intel® Xeon® W3520 processor with 8 GB of memory. Numerical results for the three experiments described above are given in Table 4.2. The tolerance for the global Gauss–Seidel was chosen so that we do not observe any visual disturbance. In our internal units, the tolerance ranges from 10^{-4} when the motion is fast to 10^{-6} in long static phases[5]. Though the solver sometimes failed to strictly reach the requested precision (in less than 2% of the cases), large errors never occur: the solver always gives an approximate solution from which the simulation can go on without exhibiting any artifact. To improve the convergence of the global algorithm, the tolerance for the local solvers is always set to a lower value, typically 10^{-7}.

Each simulation ran at a rate of a few minutes per frame, that is, 25 seconds of video in about 48 hours.

[5]This roughly translates into an average dimensionless relative error on $\boldsymbol{r}$ ranging from 10^{-3} to 10^{-4}.

Figure 4.4: *Our simulations (right) capture the complex patterns emerging from static friction in real hair (left).*

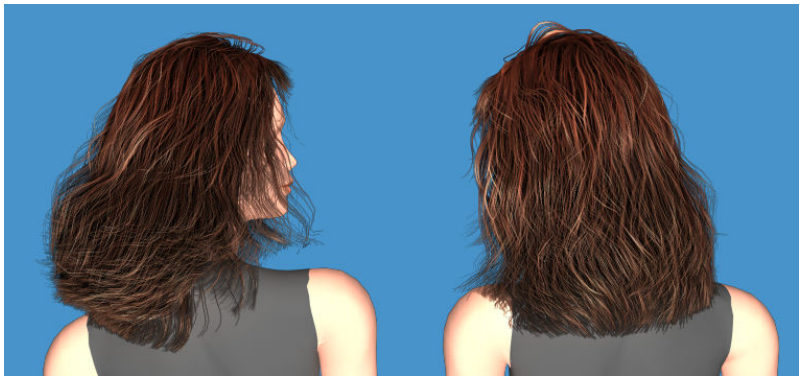


Figure 4.5: *Our method can also handle curly hair. Note how the volume of the hairstyle is preserved throughout the simulation, and how static friction is properly captured near the top of the head.*

4.2.3 Friction solvers comparisons

We compared our method against a variety of frictional contact solvers,

- **MFB**: Our Newton method based on the modified Fischer-Burmeister formulation. It can be used either as a global solver or as a local solver within a Gauss–Seidel loop.
- **PAC** and **DAC**: Respectively pure and damped Newton methods based on the Alart-Curnier formulation used in (Bertails-Descoubes et al. 2011). Both can be used as either global or local solvers.
- **Duriez08**: Local solver from Duriez (2008), which consists of successive iterations of the approach from Duriez et al. (2006) embedded in a fixed point loop. Unlike the original

approach, it solves the exact Coulomb friction problem when convergence is achieved.

- **4sides**: The local solver from Otaduy et al. (2009), which approximates the Coulomb friction cone with a four-sided pyramid.
- **Enum**: Our quartic enumerative local solver from (Bonnefon and Daviet 2011).

Exact vs faceted Coulomb friction To illustrate the influence of the choice of the local friction formulation, we created a very simple experiment: the free end of a fiber is dropped on a ball that is rotating with sinusoidal oscillations, and a non-zero friction coefficient ($\mu = 1$) is set between the ball and the fiber. Figure 4.6 (top) shows the phase plots of the free end once it has reached its periodic orbit.

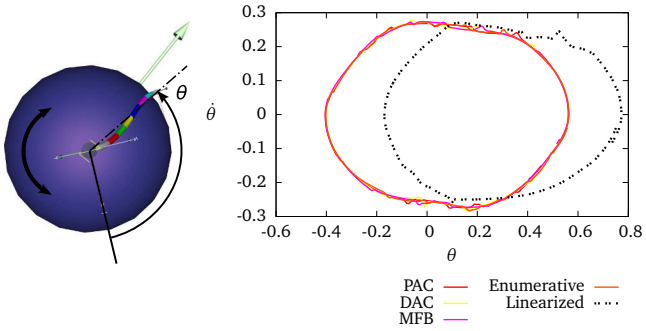


Figure 4.6: *Exact vs approximate model for Coulomb friction. Periodic orbit of the free end of a rod resting on top of a rotating sphere. Frictional contact is simulated using different solvers. Solvers that model exact Coulomb friction all reach the same orbit, no matter the choice of the error function. The linearized model (dashed lines) reaches a completely different one.*

We tried two global Newton methods, **DAC** and **MFB**, and five local solvers within a Gauss–Seidel loop: **DAC** and **MFB** again, as well as **Duriez08**, **4sides** and **Enum**. For **DAC** and **MFB** based solvers, the stopping criterion relied on the norm of the corresponding objective functions, while the other methods were stopped as soon as the step size between two Gauss–Seidel iterations got below 5%.

No matter the choice of the stopping criterion, all solvers that model exact Coulomb friction converged towards the same orbit. For the linearized local solver, (**4sides**), the Gauss–Seidel algorithm still converged quickly to a fixed point, but the trajectory differed substantially — even though as suggested by Otaduy et al. (2009), we aligned the friction pyramid with the unconstrained tangential velocity.

Performance We could not get the full-scale simulations to complete with any local solver other than our hybrid method **MFB+Enum**. All other solvers led to the divergence of the fiber model at some point of the simulation. Still, to quantitatively evaluate the performance of our solver, we saved about three hundred one-step problems from our smallest simulation, “Pony”. We then successively ran on this benchmark all the local solvers mentioned above, with or without the help of the enumerative solver.

Convergence plots are shown in Figure 4.7 and numerical results are given in Table 4.3. We used the Fischer-Burmeister error measure for all our tests, except for the pure **4sides** solver which does not attempt to solve exact Coulomb friction[6].

[6]In this case, we resorted to a measure of the iterates length. To avoid introducing any bias, we made sure that the tolerance was such that the timings of **MFB+Enum** were roughly the same under both error measures.

Local solver	$\%_{\text{alt}}$ [1]	$\%_{\text{fail}}$ [2]	$\%_{>\text{tol}}$ [3]	$\overline{\text{iters}}$ [4]	$\overline{T_{\text{GS}}}$ [6] (ms)
4sides	–	0	0	575	6497
Duriez08	–	2.45	50	267	4336
PAC	–	0.43	19.3	163	1265
DAC	–	0.26	0.33	60	874
MFB	–	0.13	4.9	72	484
Enum	–	0.0005	1	67	1044
4sides + Enum	38.7	0.001	0	47	763
Duriez08 + Enum	51	0.0007	0.33	90	1447
PAC + Enum	0.1	10^{-5}	0	62	543
DAC + Enum	0.09	10^{-6}	0	57	789
MFB + Enum	**0.07**	$\mathbf{10^{-6}}$	**0**	**41**	**312**

[1] $\%_{\text{alt}}$: Percentage of calls to fail-safe
[2] $\%_{\text{fail}}$: Percentage of local problems that did not reach tolerance
[3] $\%_{>\text{tol}}$: Percentage of one-step problems that were not solved to (global) tolerance (tol $= 10^{-6}$ except for **4sides**: 5×10^{-2})
[4] $\overline{\text{iters}}$: Mean number of Gauss–Seidel iterations
[5] $\overline{T_{\text{GS}}}$: Mean time in Gauss–Seidel algorithm

Table 4.3: *Performance comparison of various local solvers on a set of 306 one-step problems.*

From these numerical results we first note that using a linearized cone does not necessarily bring better time performance, despite a lower cost per call to the local solver. The **4sides** method was actually the one that required the highest number of Gauss–Seidel iterations to reach sufficient accuracy. We also observed that the contact freezing policy was not of much help to the **4sides** solver which, as a result, had to process a higher number of local problems per Gauss–Seidel iteration than exact friction solvers. Using a finer Coulomb friction model thus does *not necessarily* imply more costly simulations, quite the reverse in our case.

A second interesting point deals with the analysis of the role played by the **Enum** solver. For all the local solvers we tested, using the **Enum** solver as a fail-safe improved both the success rate and the computation time. In such a configuration, it turns out to be unnecessary to use a very robust primary solver. Indeed, the **MFB**, the **PAC** and more surprisingly the **4sides** method all outperformed the **DAC** solver. We also noted that the **Enum** solver requires a large number of iterations to reach the global tolerance, despite a very low rate of unsolved local problems. This is due to problems that do not admit an analytical solution, but for which an approximate solution with a low numerical error can still be found. As mentioned previously, while optimization-based solvers manage to find such acceptable solutions, the **Enum** solver remains stuck, thus spoiling the global convergence of the Gauss–Seidel algorithm. When using the **Enum** solver as a fail-safe, the rate of remaining problems without solution fortunately becomes very low, and in these rare cases, simply resetting the force to zero proved to be sufficient.

Overall, these results confirm our claim that our hybrid method **MFB+Enum** is both robust *and* efficient. The **MFB** solver alone turns out to be both faster and more robust than **PAC**, and, although slightly less robust, far much faster than **DAC**. The robustness issue becomes irrelevant when the solvers are combined with **Enum**.

Comparison of global solvers Finally, we compared our hybrid Gauss–Seidel solver to other approaches from Chapter 3 on a set of 378 DCFP coming from hair simulations with different fiber models (Super-Helices and Discrete Elastic Rods), and number of contacts ranging from 7000 to 27000. The results of this benchmark, shown on Figure 4.8, crowned Algorithm 4.1 as more efficient and robust than all the other solvers, and confirmed the speed-gain brought by the sleeping heuristics. Note also that computing the explicit Delassus operator proved to be more efficient that using the matrix-free algorithm. We also compared with two variants of the projected gradient descent algorithm, **APGD** from Heyn (2013), and **ASPG** adapted from Tasora

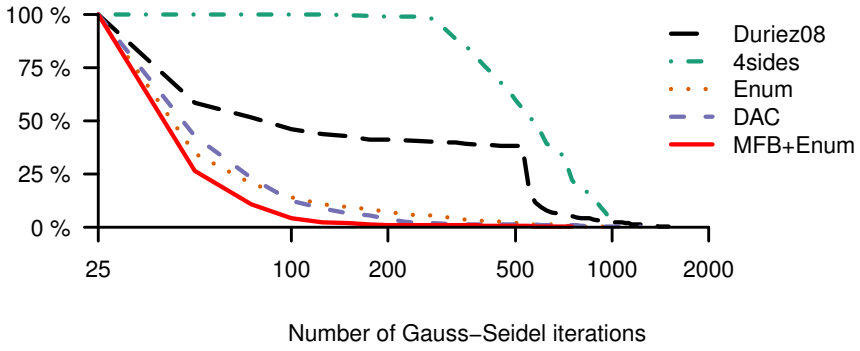


Figure 4.7: *Percentage of one-step problems (Y axis) requiring more than a given number of Gauss–Seidel iterations (X axis) to converge, for various local solvers. Our hybrid method (solid red) induces better global convergence than previous approaches.*

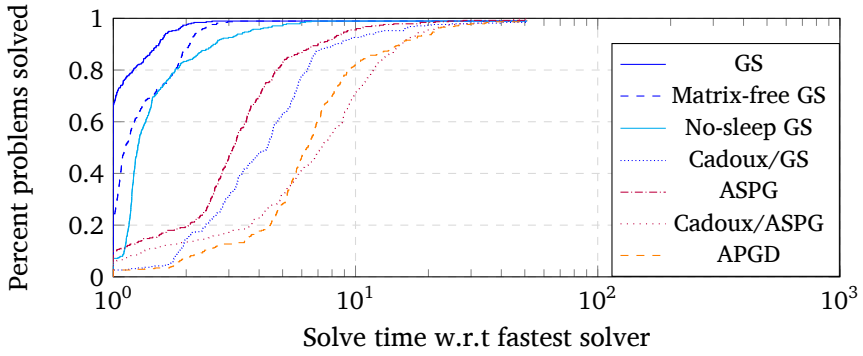


Figure 4.8: *Performance profiles for a variety of solvers on our hair simulation benchmark — percentage of problems that each solver was able to solve under a multiple x of the time taken by the fastest solver for this problem. The line $x = 1$ gives the solvers that were the most often the fastest, while $y = 1$ shows the most robust ones.*

(2013) as per Algorithm B.2. Finally, **GS** and **ASPG** were also compared to their SOCQP variants inside a Cadoux fixed-point loop.

4.2.4 Limitations

Even though our hair simulations appear more convincing than those generated by previous methods, they do not perfectly match the real movement depicted in the reference videos. This is partly due to inaccuracies in the hair styling process and to the difficulty of precisely identifying the true physical parameters. The next section will present a first step towards reducing this inaccuracy by deducing the groom rest parameters from an input geometry.

Another reason is related to the actual complexity of hair interactions. While Coulomb friction is a key ingredient, further effects such as anisotropic friction, adhesion, or electrostatics are likely to influence the hair motion, depending on the hair state (clean, dirty, wet) and external conditions (dry or wet atmosphere). Moreover, air–hair friction becomes preponderant during energetic motions, and a full simulation the surrounding air would be necessary to properly capture the intricate dynamics of this tightly coupled system.

Figure 4.9: *Our nodal algorithm efficiently handles the frictional contacts between the mannequin and her dress during a walking cycle.*

4.3 Application to cloth simulation

In this section, we depart slightly from the framework of Discrete Element Models and consider instead the simulation of a single discretized piece of cloth. Several numerical models exist for unconstrained cloth, which we will not detail here; we used an implementation of the Discrete Shell model (Grinspun et al. 2003) written by R. Casati (2015).

Treatment of cloth contacts in the Computer Graphics community typically rely upon the timestepping framework of Bridson et al. (2002), as in (Brochu et al. 2012; Harmon, Vouga, Tamstorf, et al. 2008), or event-driven asynchronous integrators (Ainsley et al. 2012; Harmon, Vouga, Smith, et al. 2009). These approaches attempt to resolve contacts by moving cloth vertices, but do not consider the increase in elastic energy associated to such displacements; friction is also treated is an explicit manner. This negatively impact the stability of the simulations.

Conversely, naive implicit treatment of contacts in cloth simulations leads to performance issues, as we explain below. Most numerical cloth models (such as the Discrete Shells) use a set of vertices as their degrees of freedom; moving a single vertex will affect the internal forces only at surrounding vertices, so the stiffness matrix will be sparse. However, applying a force anywhere will induce a displacement of each of the cloth vertices; the inverse of the stiffness matrix is dense. This means that the Delassus operator will be dense as well, and our usual hybrid Gauss–Seidel will become very inefficient. We could still use other optimization-based algorithms that do not require explicitly computing the Delassus operator, such as interior points, projected gradient descent, or ADMM, see Chapter 3). Or, as proposed by Otaduy et al. (2009), embed the friction solve inside another iterative splitting algorithm, and compute the Delassus operator using only the block-diagonal part of the stiffness matrix.

In (Daviet, Bertails-Descoubes, and Casati 2015), we propose to use a different approach and present a first step towards a more efficient and physically accurate treatment of frictional contacts in cloth simulations. Note that this method suffer from severe limitations, making it totally unsuitable for real-world usage; indeed, we consider only contacts between cloth vertices and external objects (such as the character body), and assume that there is at most one contact per vertex. However, the simplicity of the formulation might be of interest, and could potentially be extended to more useful settings.

4.3.1 Nodal algorithm

Consider the DCFP for a 3D system whose degrees of freedom are its m vertices, and n contacts with external objects occurring exactly at the vertices. The relative velocities are simply given by

$$\boldsymbol{u}_i = E_i^{\mathsf{T}}(\boldsymbol{v}_j - \boldsymbol{w}_i^{ext}), \quad 1 \leq i \leq n,\ 1 \leq j \leq m$$

where j is the index of the contacting vertex at the i^{th} contact point, $\boldsymbol{w}_i^{ext}$ the velocity of the external object and $E_i := (\boldsymbol{n}_i, \boldsymbol{t}_i, \boldsymbol{n}_i \wedge \boldsymbol{t}_i)$ a rotation matrix transforming the local contact basis coordinates into the world coordinates.

The matrix H then possesses a very simple structure: there is only one non-zero block per contact, $H_{i,j} = E_i^{\mathsf{T}}$. H is non invertible; however, assuming that there is at most one contact per

vertex, we can easily extend it into a square $3m \times 3m$ block-diagonal matrix G,

$$G_{i,i} := \begin{cases} E_i^\intercal & \text{if vertex } i \text{ is in contact} \\ \mathbb{I} & \text{otherwise .} \end{cases} \tag{4.15}$$

The DCFP can then be equivalently written using this matrix G as finding $\boldsymbol{v}, \boldsymbol{u}, \boldsymbol{r}$, all in $\mathbb{R}^{3m}$, such that

$$\begin{cases} M\boldsymbol{v} = \boldsymbol{f} + G^\intercal \boldsymbol{r} \\ \boldsymbol{u} = G\boldsymbol{v} - G\boldsymbol{w}^{ext} \\ (\boldsymbol{u}_i, \boldsymbol{r}_i) \in \mathscr{C}_{\mu_i} & \text{if vertex } i \text{ is in contact} \\ \boldsymbol{r}_i = \boldsymbol{0} & \text{otherwise .} \end{cases} \tag{4.16}$$

Now, G is orthonormal, and can be easily inverted as $G^{-1} = G^\intercal$. We can thus eliminate $\boldsymbol{v}$ in system (4.16) and write its first line as

$$MG^\intercal \left(\boldsymbol{u} + \boldsymbol{w}^{ext}\right) = \boldsymbol{f} + G^\intercal \boldsymbol{r}.$$

Multiplying both sides by G, we obtain system (4.17)

$$\begin{cases} \underbrace{GMG^\intercal}_{\tilde{W}} \boldsymbol{u} + \underbrace{G(MG^\intercal \boldsymbol{w}^{ext} - \boldsymbol{f})}_{\tilde{\boldsymbol{b}}} = \boldsymbol{r} \\ (\boldsymbol{u}_i, \boldsymbol{r}_i) \in \mathscr{C}_{\mu_i} & \text{if vertex } i \text{ is in contact} \\ \boldsymbol{r}_i = \boldsymbol{0} & \text{otherwise .} \end{cases} \tag{4.17}$$

System (4.17) is very similar to the dual formulation of the DCFP (2.19), where the role of $\boldsymbol{u}$ and $\boldsymbol{r}$ have been reversed. However, the matrix $\tilde{W}$ is now sparse, and we can therefore solve (4.17) using our Gauss–Seidel algorithm by either

- transforming (4.17) as a sequence of convexified DCFP, using a variation of the Cadoux fixed-point algorithm (see Daviet, Bertails-Descoubes, and Casati 2015), or;
- if $\tilde{W}_{ii}$ is invertible, writing the inverse local linear relationship, $\boldsymbol{u}_i = \tilde{W}_{ii}^{-1}(\boldsymbol{r}_i - \tilde{\boldsymbol{b}}_i)$ and using our original one-contact solvers, or;
- devising a new hybrid local solver for Coulomb friction with reversed linear relationship between the force and relative velocity. The Fischer–Burmeister Newton algorithm can be trivially adapted, and a corresponding quartic polynomial has been derived by a student in our group, L. Toran — but has not been published yet.

One can also take advantage of the orthonormality of the matrix G by performing a projected gradient descent algorithm on the primal formulation. Indeed, the feasible set of each intermediate primal SOCQP of the Cadoux algorithm is

$$\begin{aligned} C(\boldsymbol{s}) &= \{\boldsymbol{v} \in \mathbb{R}^{3m}, G\boldsymbol{v} + \boldsymbol{w}^{ext} + \boldsymbol{s} \in K := K_1 \times \ldots \times K_m\} \\ K_i &:= \begin{Bmatrix} \mathscr{K}_{\frac{1}{\mu_i}} & \text{if } i \text{ is in contact} \\ \mathbb{R}^d & \text{otherwise} \end{Bmatrix} \qquad \forall 1 \leq i \leq m \end{aligned}$$

and the projection on $C(\boldsymbol{s})$ is now easy to compute:

$$\begin{aligned} \boldsymbol{y} = \Pi_{C(\boldsymbol{s})}(\boldsymbol{v}) &\iff \boldsymbol{v} - \boldsymbol{y} \in \mathscr{N}_{C(\boldsymbol{s})}(\boldsymbol{y}) \\ &\iff \langle \boldsymbol{v} - \boldsymbol{y}, \boldsymbol{z} - \boldsymbol{y} \rangle \leq 0 && \forall \boldsymbol{z} \in C(\boldsymbol{s}) \\ &\iff \left\langle \boldsymbol{v} - \boldsymbol{y}, G^\intercal \left(\boldsymbol{u} - \boldsymbol{s} - \boldsymbol{w}^{ext}\right) - \boldsymbol{y} \right\rangle \leq 0 && \forall \boldsymbol{u} \in K \\ &\iff \left\langle G\boldsymbol{v} - G\boldsymbol{y}, \boldsymbol{u} - \left(G\boldsymbol{y} + \boldsymbol{s} + \boldsymbol{w}^{ext}\right) \right\rangle \leq 0 && \forall \boldsymbol{u} \in K \\ &\iff (G\boldsymbol{y} + \boldsymbol{s} + \boldsymbol{w}^{ext}) \in \mathscr{N}_K(G\boldsymbol{v} + \boldsymbol{s} + \boldsymbol{w}^{ext}) \\ &\iff \boldsymbol{y} = G^T \left(\Pi_K \left(G\boldsymbol{v} + \boldsymbol{s} + \boldsymbol{w}^{ext}\right) - \boldsymbol{s} - \boldsymbol{w}^{ext}\right). \end{aligned}$$

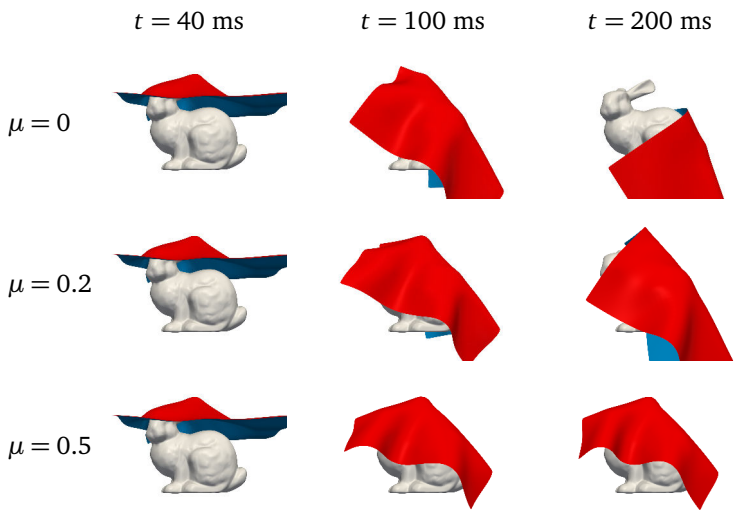


Figure 4.10: *When the friction coefficient μ is high enough (bottom), the cloth falling onto the bunny no longer slides away.*

Scene	m^1	n^2	Dual Build	Dual Solve	Nodal algorithm Build	Nodal algorithm Solve
Bunny	6500	169	2.25	0.017	0.02	0.074
Skirt	5208	741	76.7	0.23	0.074	0.14
Dress	6060	1086	157	0.19	0.077	0.048

Times are in seconds.
[1] Number of cloth vertices.
[2] Average number of contacts.

Table 4.4: *Performance comparison between the standard dual algorithm (with explicit Delassus operator) and the nodal algorithm for solving body–cloth contacts.*

4.3.2 Results

Capturing dry friction We first want to ensure that our algorithm is indeed capable of capturing the effects of Coulomb friction. We run a simple simulation consisting of a square piece of cloth falling onto the Stanford bunny (Figure 4.10). While the friction coefficient is low, the cloth is slipping down under its own weight; however, for higher values of this coefficient, friction maintains the cloth stuck atop the bunny.

Performance We compared the cost of solving the DCFP using the standard (dual) Gauss–Seidel algorithm with the dense Delassus operator versus running our nodal algorithm (using the Cadoux fixed-point algorithm and our hybrid Gauss–Seidel for each intermediate SOCQP). Table 4.4 summarize our results for three test cases: the cloth-on-bunny simulation of Figure 4.10, the walk of a mannequin wearing a dress of as in Figure 4.9), and a similar motion with a skirt. When taking into account the cost of computing the Delassus operator, the nodal algorithm ran about two orders of magnitude faster than the naive dual method.

4.3.3 Limitations

Our naive nodal algorithm allowed for a significant speed-up w.r.t. solving the standard DCFP on simple simulations, such as that of Figure 4.9 where the dress is contacting only with the underlying body . As the body mesh does not have sharp features, dealing only with contacts

at cloth vertices did not prove to be too limiting. However, the inability to account for self-contacts becomes quite disturbing when folds overlap with each other during motion. Yet, future work could probably accommodate for self-contacts while retaining the efficiency of the nodal algorithm. Indeed, the relative velocity at a self-contacting point between two vertices reads $\boldsymbol{u}_i = \boldsymbol{v}_j - \boldsymbol{v}_k$, and the correspond block-row of H, $H_{i\cdot}$, contains only two blocks, $H_{i,j} = E_i^\intercal$ and $H_{i,k} = -E_i^T$. Now, as the Coulomb law is invariant w.r.t. a constant scaling factor on $\boldsymbol{u}_i$, this row could be replaced with $H_{i,j} = \frac{1}{\sqrt{2}} E_i^\intercal$ and $H_{i,k} = -\frac{1}{\sqrt{2}} E_i^T$. Then, assuming that the vertices j an k are not involved in any other contact, an orthonormal matrix G can still be constructed by defining its j^{th} and k^{th} rows as

$$G_{j,j} := \frac{1}{\sqrt{2}} E_i^\intercal \qquad G_{j,k} = -\frac{1}{\sqrt{2}} E_i^T$$
$$G_{k,j} := \frac{1}{\sqrt{2}} E_i^\intercal \qquad G_{k,k} = \frac{1}{\sqrt{2}} E_i^T.$$

The j^{th} and k^{th} constraints should then be $(\boldsymbol{u}_j, \boldsymbol{r}_j) \in \mathscr{C}_{\mu_j}$ and $\boldsymbol{r}_k = \boldsymbol{0}$, respectively.

However, this still does not address all of the limitations of our approach. The constraint of having only contacts at cloth vertices, and at most one contact per vertex is severe, though it might be alleviated by adaptively spawning vertices with kinematic constraints. Furthermore, cloth are thin objects. In contrast to 3D bodies, for which it is easy to determine what is outside and what is inside, if penetration between two layers of cloth is not resolved at one time step, it will no longer be possible in the following ones to know which layer should be on "top" of the other. Our velocity-based method does not provide this guarantee, in contrast to, for instance, (Brochu et al. 2012).

4.4 Inverse modeling with frictional contacts

Until now, we were interested in the direct simulation of mechanical systems; that is, given a discrete set of physical parameters $\boldsymbol{p} \in \mathbb{R}^p$ describing the system (for instance, the Young modulus of the material, friction coefficients, etc), and initial positions and velocities, we wanted to compute the trajectory of the system.

Inverse modeling consists in the reverse problem: given the trajectory of the system, can we get back to its physical parameters ? In this section, we present briefly two inverse modeling problems which have the particularity of being again able to be expressed as a (or a sequence of) SOCQP, and thus are good candidates for the application of our Gauss–Seidel algorithm. These works have been spearheaded by A. Derouet-Jourdan for fiber inversion and R. Casati for cloth inversion, and have already been treated in much more details in their respective dissertations (Casati 2015; Derouet-Jourdan 2013).

We consider a mechanical system at equilibrium (i.e., $\boldsymbol{v} = \boldsymbol{0}$) under internal, external and unilateral frictional contact forces. Writing that the sum of all forces must be zero, there holds

$$\boldsymbol{0} = \boldsymbol{f}^{\text{int}}(\bar{\boldsymbol{q}}; \boldsymbol{p}) + \boldsymbol{f}^{\text{ext}}(\bar{\boldsymbol{q}}) + H^\intercal(\bar{\boldsymbol{q}})\boldsymbol{r}, \tag{4.18}$$

where $\bar{\boldsymbol{q}}$ denotes the generalized coordinates of the system at equilibrium, and the unknowns are $\boldsymbol{p}$ and $\boldsymbol{r}$ such that $\forall i = 1 \dots n$, $\boldsymbol{r}_i \in \mathscr{K}_{\mu_i}$.

4.4.1 Linear case

We can first assume that the internal forces derive from a potential energy that is quadratic in $\boldsymbol{p}$. This is the case treated in (Derouet-Jourdan, Bertails-Descoubes, Daviet, et al. 2013), where we attempt to retrieve the natural curvatures of a set of Super-Helices (Bertails et al. 2006), assuming that the fibers' Young modulus, radii, volumetric masses and friction coefficients are known.

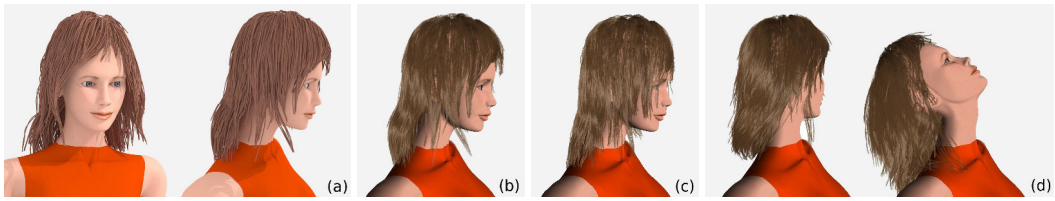


Figure 4.11: *Outline of our hair inverse modeling strategy.* ***(a)*** *Input hair geometry, here a capture from Herrera et al. (2012).* ***(b)*** *Conversion to a mechanical fiber model, here a set of Super-Helices.* ***(c)*** *Without inversion, the hair sags at the beginning of the simulation, and the groom is lost.* ***(d)*** *Using our inversion procedure, the groom stays in place and behaves naturally once the head starts moving.*

This addresses an important problem in the visual effects industry: when creating a digital double of a live performing actor, artists (or sufficiently-advanced software) will sculpt the groom so that it matches a reference photography. One can "easily" transform each of the groom's strands into the rest configuration of an animatable fiber (e.g., a Super-Helix, as in Derouet-Jourdan, Bertails-Descoubes, and Thollot 2013). However, once the simulation is run with gravity, the fibers will fall from their rest configuration and significant sagging will be observed, ruining the similarity between the digital groom and its physical reference (Figure 4.11). One could simply consider fibers one by one, and find a set of rest curvatures and twists so that their deformed configuration under gravity matches the input groom (see e.g., Derouet-Jourdan, Bertails-Descoubes, and Thollot 2010). Then, no sagging will occur when the simulation is run. However, such an inversion procedure will explain all deformations, including those due to contacts (such as hair resting upon a shoulder) as merely stemming from large rest curvatures, which will inevitably lead to weird dynamics. To avoid such artifacts, one must consider the whole groom and take contacts into account when looking for the rest parameters.

Inversion strategy In the linear case, the inverse problem (4.18) boils down to finding $\boldsymbol{p} \in \mathbb{R}^p$ and $\boldsymbol{r} \in \mathbb{R}^{nd}$ such that

$$\mathbf{0} = \boldsymbol{f}^{\text{ext}} + K\boldsymbol{p} + H^\intercal \boldsymbol{r}, \qquad \boldsymbol{r} \in \Pi \mathscr{K}_{\mu_i}. \tag{4.19}$$

Even when K is symmetric positive definite, problem (4.19) is under-determined; we can always find a solution for which $\boldsymbol{r} = \mathbf{0}$, but this might not be the only one.

To circumvent this problem, we define a target parameter set $\boldsymbol{p}^0 \in \mathbb{R}^p$ from physical considerations, and minimize the distance between our retrieved parameter $\boldsymbol{p}$ and $\boldsymbol{p}^0$. Our problem becomes

$$\begin{aligned}\min_{\boldsymbol{p}\in\mathbb{R}^p} \frac{1}{2}\|\boldsymbol{p}-\boldsymbol{p}^0\|^2 &= \min_{\boldsymbol{r}\in\Pi\mathscr{K}_{\mu_i}} \frac{1}{2}\|K^{-1}\left(H^\intercal\boldsymbol{r}+\boldsymbol{f}^{\text{ext}}\right)+\boldsymbol{p}^0\|^2 \\ &= \min_{\boldsymbol{r}\in\Pi\mathscr{K}_{\mu_i}} \frac{1}{2}\boldsymbol{r}^\intercal W\boldsymbol{r} + \boldsymbol{r}^T\boldsymbol{b} + \frac{1}{2}\|K^{-1}\boldsymbol{f}^{\text{ext}}+\boldsymbol{p}^0\|,\end{aligned}$$

with $W = HK^{-2}H^\intercal$ and $\boldsymbol{b} = H\left(K^{-2}\boldsymbol{f}^{ext} + K^{-1}\boldsymbol{p}^0\right)$, whose optimality conditions satisfy a problem structurally similar to the convexified DCFP (2.22),

$$\begin{cases} K^2\boldsymbol{v} = \boldsymbol{f}^{ext} + K\boldsymbol{p}^0 + H^\intercal\boldsymbol{r} \\ \boldsymbol{u} = H\boldsymbol{v} \\ \mathscr{K}_{\frac{1}{\mu_i}} \ni \boldsymbol{u}_i \perp \boldsymbol{r}_i \in \mathscr{K}_{\mu_i} \qquad \forall i = 1\dots n. \end{cases} \tag{4.20}$$

As we saw in Section 2.3.2, problem (4.20) always admits a solution in $\boldsymbol{v}$ and can be solved either as a primal or dual SOCQP. $\boldsymbol{p}$ can then be deduced from a solution to (4.20) as $\boldsymbol{p} = -K^{-1}\left(H^\intercal\boldsymbol{r} + \boldsymbol{f}^{\text{ext}}\right) = \boldsymbol{p}^0 - K\boldsymbol{v}$.

(a) (b) (c) (d)

Figure 4.12: *An application of inverse cloth modeling.* ***(a)*** *An artist creates a skirt model by sculpting its rest pose under gravity.* ***(b)*** *Without inverse modeling, the simulated cloth sags and falls from the hips of the character.* ***(c)*** *Inverse modeling ensures that friction at the hips suffices to maintain the cloth at its original position.* ***(d)*** *The recovered cloth behaves naturally when external forces (wind) are applied.*

Application to inverse hair modeling The above framework for taking frictional contacts into account within the inversion procedure requires defining a set of target parameters $\boldsymbol{p}^0$, and three possibilities can be considered:

- Using directly the input curvatures from the deformed configurations;
- Using the curvature computed at the end of each input curve; this assumes that each fiber possesses relatively constant natural curvatures, and that the fiber's end is the least-deformed portion.
- Using a vanishing target curvatures, assuming that the fibers are naturally straight.

Each of these strategies yields a different balance between the influence of the contact forces and that of the natural curliness, and we have not found any of them to work better than the others overall. Instead, we suggest to try each of them for any input, and manually compare the results.

The resulting SOCQP will also often have a singular W matrix, which means underdetermination of the contact forces. In practice, we propose to regularize the minimization problem by adding a penalization term $\varepsilon \mathbb{I}_{\mathbb{R}^{nd}}$ to W, preventing the inversion procedure to consider unlikely huge contact forces. This strategy allowed us to find plausible parameters for input geometries with hundreds of fibers and thousands of contacts, at a cost similar to that of a single simulation step. The hair recovered from the inversion procedure could then either be simulated unmodified, or further edited using physically-based grooming techniques.

4.4.2 Nonlinear case

In (Casati et al. 2016), we proposed a first attempt to solve the inverse problem (4.18) with a nonlinear dependency of the forces w.r.t. the parameters $\boldsymbol{p}$. This is the case for the Discrete Elastic Shell model (Grinspun et al. 2003), when the parameters to recover are the positions of the vertices of the rest shape.

We assume that all non-contact forces derive from a potential energy $\mathscr{E}^p(\boldsymbol{q},\boldsymbol{p})$, i.e.,

$$\boldsymbol{f}^{int}(\boldsymbol{q};\boldsymbol{p}) + \boldsymbol{f}^{ext}(\boldsymbol{q}) = -\frac{\partial \mathscr{E}^p}{\partial \boldsymbol{q}}(\boldsymbol{q},\boldsymbol{p}).$$

We introduce the equilibrium function $F : \mathbb{R}^m \times \mathbb{R}^p \to \mathbb{R}^m$,

$$F : (\boldsymbol{q},\boldsymbol{p}) \mapsto \frac{\partial \mathscr{E}^p}{\partial \boldsymbol{q}}(\boldsymbol{q},\boldsymbol{p}) - \Pi_C\left(\frac{\partial \mathscr{E}^p}{\partial \boldsymbol{q}}(\boldsymbol{q},\boldsymbol{p})\right)$$

where $C = H(\bar{\boldsymbol{q}})^\intercal \Pi \mathscr{K}_{\mu_i}$, such that

$$F(\boldsymbol{q},\boldsymbol{p}) = \boldsymbol{0} \iff \frac{\partial \mathscr{E}^p}{\partial \boldsymbol{q}}(\boldsymbol{q},\boldsymbol{p}) \in H(\bar{\boldsymbol{q}})^\intercal \Pi \mathscr{K}_{\mu_i}.$$

The inverse problem (4.18) is thus equivalent to finding $\boldsymbol{p}$ such that $F(\bar{\boldsymbol{q}},\boldsymbol{p}) = \boldsymbol{0}$. Just like for the dynamics (Section 4.3), we assume that the degrees of freedom $\boldsymbol{q}$ of the dynamical model are the positions of the shell vertices, and that contacts occur only at those points. This means that the projection on C simply amounts to projections on rotated SOC, and is thus trivial to compute.

Draping function Without diving too deeply into the algorithmic details, we then recast (4.18) as the minimization problem

$$\min_{\boldsymbol{p}\in\mathbb{R}^p} J(\boldsymbol{p}) := \frac{1}{2}\|\Phi(\boldsymbol{p}) - \bar{\boldsymbol{q}}\|^2 \tag{4.21}$$

where $\Phi : \mathbb{R}^p \to \mathbb{R}^m$ is a so-called *draping* function, which associates to a parameter $\boldsymbol{p}$ an equilibrium coordinate $\boldsymbol{q}$, i.e., such that $F(\boldsymbol{q},\boldsymbol{p}) = \boldsymbol{0}$. For instance, Φ can be defined locally through the implicit function theorem[7]. One solution to evaluate Φ would be to perform a full simulation, but that would be too costly. Instead, let us now show how we can evaluate Φ as the solution to an optimization problem.

As $\forall \boldsymbol{x} \in \mathbb{R}^3, -\mathscr{N}_{\mathscr{K}_{\frac{1}{\mu}}}(\boldsymbol{x}) \subset \mathscr{K}_\mu$, a sufficient condition to obtain $F(\boldsymbol{q},\boldsymbol{p}) = \boldsymbol{0}$, and thus $\boldsymbol{q} = \Phi(\boldsymbol{p})$, is to have $\frac{\partial \mathscr{E}^p}{\partial \boldsymbol{q}}(\boldsymbol{q},\boldsymbol{p}) \in -H(\bar{\boldsymbol{q}})^T \mathscr{N}_{\Pi\mathscr{K}_{\frac{1}{\mu_i}}}(H(\bar{\boldsymbol{q}})(\boldsymbol{q}-\bar{\boldsymbol{q}}))$; the condition also becomes necessary when $\Phi(\boldsymbol{p}) = \bar{\boldsymbol{q}}$, i.e., when $\boldsymbol{p}$ is a solution to the inverse problem. Under our nodal contact assumption $H(\bar{\boldsymbol{q}})$ is surjective, so we can apply the Corollary A.3 to Property A.12 on the subdifferential of the precomposition with an affine map and obtain an equivalent condition,

$$\frac{\partial \mathscr{E}^p}{\partial \boldsymbol{q}}(\boldsymbol{q},\boldsymbol{p}) \in -\mathscr{N}_V(\boldsymbol{q}) \tag{4.22}$$

with $V = \left\{\boldsymbol{q} \in \mathbb{R}^m,\ H(\boldsymbol{q}-\bar{\boldsymbol{q}}) \in \Pi\mathscr{K}_{\frac{1}{\mu_i}}\right\}$. We recognize this last inclusion as defining a (local) minimum of an optimization problem,

$$\min_{\boldsymbol{q}\in V} \mathscr{E}^p(\boldsymbol{q},\boldsymbol{p}). \tag{4.23}$$

Evaluating the draping function $\Phi(\boldsymbol{p})$ therefore amounts to finding a local minimum of an optimization problem under SOC constraints, which can be cast as a sequence of convexified DCFP:

$$\begin{cases} \dfrac{\partial^2 E_c}{\partial \boldsymbol{q}^2}(\boldsymbol{q}^k,\boldsymbol{p})\left(\boldsymbol{q}^{k+1}-\boldsymbol{q}^k\right) = -\dfrac{\partial E_c}{\partial \boldsymbol{q}}(\boldsymbol{q}^k,\boldsymbol{p}) + H(\bar{\boldsymbol{q}})^\intercal \boldsymbol{r} \\ \boldsymbol{u} = H(\bar{\boldsymbol{q}})(\boldsymbol{q}^{k+1}-\bar{\boldsymbol{q}}) \\ \mathscr{K}_{\frac{1}{\mu_i}} \ni \boldsymbol{u}_i \perp \boldsymbol{r}_i \in \mathscr{K}_{\mu_i} \qquad\qquad \forall i = 1\dots n. \end{cases} \tag{4.24}$$

In practice, we penalize the objective function with a proximal regularization term $\frac{1}{2\lambda}\|\boldsymbol{q}-\bar{\boldsymbol{q}}\|^2$ in order to ensure that the Hessian in (4.24) remains reasonably positive, and avoid finding local minima that are too far away from the target position. λ is then progressively increased as the overall algorithm converges.

[7] This definition does not hold in the presence of contacts, as F is no longer differentiable, but we can still define Φ algorithmically.

Gradient descent Now that we know how to compute $\Phi(\boldsymbol{p})$, let us go back to our original minimization problem (4.21). Using a gradient-descent algorithm requires evaluating the gradient of J, $\frac{\partial J}{\partial \boldsymbol{p}}(\boldsymbol{p}) = 2\left(\frac{\partial \Phi}{\partial \boldsymbol{p}}(\boldsymbol{p})\right)^{\intercal}(\Phi(\boldsymbol{p})-\bar{\boldsymbol{q}})$. $\frac{\partial \Phi}{\partial \boldsymbol{p}}$ can be evalulated from the application of the implicit function theorem (see Casati et al. 2016 for more details) on the local optimality condition of (4.23), Equation (4.22), formulated as an orthogonal projection as per Corollary A.6,

$$\mathbf{0} = \Pi_V\left(\boldsymbol{q} - \frac{\partial \mathscr{E}^p}{\partial \boldsymbol{q}}(\boldsymbol{q},\boldsymbol{p})\right) - \boldsymbol{q}.$$

Results We attempted to use Algorithm 4.1 to evaluate the draping function, but obtained mitigated results. Indeed, as the stiffness matrix of the intermediate DCFP (4.24) can still have negative eigenvalues, our hybrid Gauss–Seidel algorithm does not always achieve convergence. The non-linear projected gradient descent algorithm presented in (Casati et al. 2016) proved to be more robust (though less efficient), and was used as a fail-safe. Evaluating the draping function thus constituted a major computational bottleneck of the inversion procedure.

For the global optimization process, we considered gradient descent, conjugate gradient descent, and L-BFGS strategies. Achieving convergence with any of those strategies required to equip them with a line search, and thus demanded a very high number of evaluations of the draping function. This limited the applicability of the method to simple examples for which the rest shape was not to far from the target. However, our method was still able to yield useful results, such as preventing a skirt from falling from the hips of a character, as illustrated on Figure 4.12.

Discussion

In this chapter, we presented a simple variation of a classical algorithm which performed efficiently and robustly over a variety of applications: hair dynamics and statics, cloth dynamics, and more generally, any DEM contact problem yielding a DCFP or convexified DCFP for which the inverse stiffness matrix M^{-1} is sparse. Furthermore, we have seen that some inverse modeling problems under frictional contacts also involved SOCQP, justifying again the need for efficient resolution methods.

Performance It may seem strange that an algorithm as naive as the Gauss–Seidel, subject to a lot of theoretical limitations, performs so well in practice. One explanation would be that a lot of domain knowledge is built into the local solver, especially when an analytical one is used. This is not as much the case for projective approaches, in which the constraint is only defined through an orthogonal projection operator. Note that Mazhar et al. (2015) argue that the Gauss–Seidel algorithm performs much more poorly than their APGD algorithm. However, they use a crippled variant of the Gauss–Seidel algorithm in which the local solver consist only in performing one projection of the forces onto the admissible cone, as in (Heyn 2013, p. 44). As each contact is also solved with a good precision, even if the asymptotic convergence of whole algorithm is slow, the local force values are never too far from an acceptable solution — this makes the Gauss–Seidel very robust in practice.

That being said, for now we have only tested our solver on hair examples consisting of a few thousands fibers only — that is, about 50 times smaller than a real human hair. Although we feel our solver could possibly resist some further scaling up (using a similar strategy, Kaufman, Tamstorf, et al. (2014) simulated one order of magnitude more fibers), we know that computational time will eventually become the main bottleneck, and concurrent algorithms appear to remain necessary for going beyond that.

The Gauss–Seidel algorithm may also not be the most adequate when considering smaller problems that must be solved to higher precisions, such as those of the FCLIB[8] benchmark (Acary,

[8] http://fclib.gforge.inria.fr/

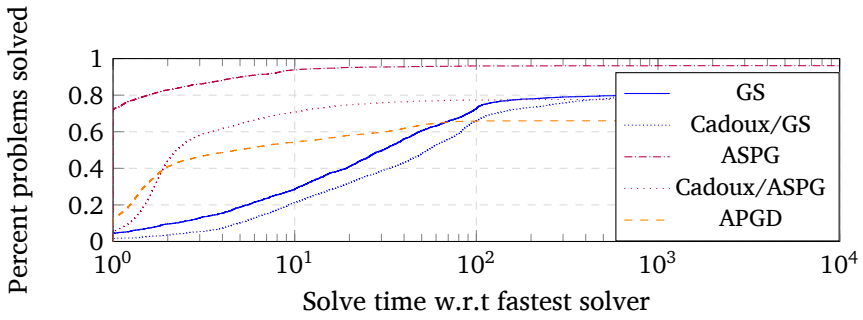


Figure 4.13: *Performance profiles for a variety of solvers on the FCLIB benchmark — percentage of problems that each solver was able to solve under a multiple x of the time taken by the fastest solver for this problem. The line $x = 1$ gives the solvers that were the most often the fastest, while $y = 1$ shows the most robust ones.*

Brémond, et al. 2014). The FCLIB-0.2 collection is a set of thousands of DCFP in reduced formulation (2.19) and without linear constraints, generated by four different kinds of simulations (stacks of rigid boxes, unstructured capsules and rigid-body chains). All problems feature less than on thousand degrees of freedom and contact points. Figure 4.13 shows performance profiles for different solvers on an arbitrary subset of 2500 problems. On this benchmark, the projected gradient variants, and especially **ASPG** (Algorithm B.2), performed much better than our Gauss–Seidel algorithm. This in part due to the fact that the combination of sleeping heuristics and naive parallelization may hamper convergence on problems with low number of contacts, as the algorithm becomes more alike to Jacobi.

Continuum mechanics Alternatively, one may renounce attempting to solve for each frictional contact force, and instead simulate the macroscopic interactions between the fibers using a continuum model. This is the strategy that we are going to adopt in the second part of this dissertation, though for much simpler systems consisting only of rigid spherical grains. While DCFP solvers (Chapter 3) have been initially devised to work on problems arising from contacts between discrete bodies (Chapter 2), we will see that, exploiting the similarity between the Coulomb and non-associated Drucker–Prager laws (Chapter 1), they can actually be used for a wider range of applications, including some problems from continuum mechanics. In some cases, we will once again be able to leverage the hybrid Gauss–Seidel algorithm presented in this chapter.

Part II

Continuum simulation of granular materials

5 *Continuum simulation of granular flows*

We already mentioned in our introduction that several attempts have been made to model the averaged behavior of granular materials. In particular, we mentioned the Mohr–Coulomb and Drucker–Prager yield surfaces, which restrict the set of admissible stresses inside granulars to take into account that:

- Granular materials offer a much higher resistance to compression than to stretching. In fact, materials like dry sand can freely dilate, but their compression Young modulus is very high (hundreds of MPa).
- Due to the Coulombic nature of the friction between grains, the resistance of granular materials to shearing increases with the local pressure.

In the following, we review a few methods that have been used in the literature to perform numerical simulations of granular materials modeled as continua.

Notation We will generally denote *values* will italic Greek (usually for tensors and scalars) or roman (for vectors and scalars) letters, and *fields* with the corresponding upright symbols. For instance, we shall write $\boldsymbol{u} = \mathbf{u}(\boldsymbol{x}, t)$ the value of the velocity field at $\boldsymbol{x} \in \Omega$ and at instant t, and $\boldsymbol{\sigma} = \boldsymbol{\sigma}(\boldsymbol{x}, t)$ the value of the stress field.

5.1 Continuum models

As the numerical simulation of granular continua has found a wide range of applications, from the small-strain study of soils to the prediction of the run-out of avalanches, it is natural that different point of views have been taken in the literature.

Most of the contitutive models devised for this purpose fit in the framework of Implicit Standard Materials (ISM) presented in Chapter 1, and boil down to (omitting potential hardening variables),

$$\begin{cases} \dot{\boldsymbol{\varepsilon}} = \dot{\boldsymbol{\varepsilon}}_e + \dot{\boldsymbol{\varepsilon}}_p & (5.1) \\ \boldsymbol{\sigma} \in \rho \dfrac{\partial \mathscr{E}}{\partial \boldsymbol{\varepsilon}_e}(\boldsymbol{\varepsilon}_e) & (5.2) \\ \boldsymbol{\sigma} \in \dfrac{\partial b}{\partial \dot{\boldsymbol{\varepsilon}}_p}(\dot{\boldsymbol{\varepsilon}}_p, \boldsymbol{\sigma}), & (5.3) \end{cases}$$

where $\mathscr{E}$ is the free-energy function and b is a bipotential (Definition 1.2). In the case of Generalized Standard Materials (GSM), b is simply defined through a dissipation potential $\mathscr{D}$ as $b(\dot{\boldsymbol{\varepsilon}}_p, \boldsymbol{\sigma}) := \mathscr{D}(\dot{\boldsymbol{\varepsilon}}_p) + \mathscr{D}^\star(\boldsymbol{\sigma})$. However, different bodies of research have considered different expressions for $\mathscr{E}$ and b in the context of granular continua.

A first class of methods (*flow-oriented*) is motivated by the liquid regime of granulars, thus the modeling of very large deformations, and derive from the standard framework of yield stress fluids. The grains are generally assumed to be perfectly rigid, so that the material is inelastic, leading to visco-plastic models.

Another class of methods (*elasticity-oriented*) has been historically focused on the study of soils and the solid behavior of granulars, in the fashion of the elasto-plastic "Cam Clay" model (Vermeer 1998).

5.1.1 Inelastic yield-stress fluids

The flow-oriented approach consists in adapting the standard framework of inelastic yield-stress fluids to account for the pressure-dependency observed in granular materials.

Incompressible yield stress fluids Fluids such as toothpaste behave like a solid at rest, but can flow once an external load is applied. Like Newtonian fluids, they remain inelastic and incompressible. In the framework of GSM, this means that the free-energy function is chosen as $\mathscr{E} = \mathscr{I}_{\{\mathbf{0}_{S_d}\}}$, so that Equation (5.2) implies $\boldsymbol{\varepsilon}_e = \mathbf{0}$. Moreover, the dissipation potential must be such that $\mathscr{D}(\gamma\mathbb{I}) = \mathscr{I}_{\{0\}}(\gamma)$, so that Equation (5.3) enforces $\operatorname{Tr}\dot{\boldsymbol{\varepsilon}} = 0$, i.e., $\nabla\cdot\mathbf{u} = 0$.

However, in contrast to Newtonian fluids, the dissipation potential is nonsmooth w.r.t. the tangential strain $\operatorname{Dev}\dot{\boldsymbol{\varepsilon}}$. The simplest constitutive law for incompressible yield-stress fluid is the Bingham rheology, of which we already presented an inviscid version in Section 0.3.1. This rheology derives from a simple yet nonsmooth visco-plastic dissipation potential $\mathscr{D}_{\text{Bi}}$,

$$\mathscr{D}_{\text{Bi}}(\operatorname{Dev}\dot{\boldsymbol{\varepsilon}}) := \eta(\operatorname{Dev}\dot{\boldsymbol{\varepsilon}})^2 + \sigma_S|\operatorname{Dev}\dot{\boldsymbol{\varepsilon}}|,$$

and can be written using a disjunctive formulation as

$$\begin{cases} \operatorname{Dev}\boldsymbol{\sigma} = \eta\operatorname{Dev}\dot{\boldsymbol{\varepsilon}} + \sigma_S\dfrac{\operatorname{Dev}\dot{\boldsymbol{\varepsilon}}}{|\operatorname{Dev}\dot{\boldsymbol{\varepsilon}}|} & \text{if } \operatorname{Dev}\dot{\boldsymbol{\varepsilon}} \neq \mathbf{0} \\ |\operatorname{Dev}\boldsymbol{\sigma}| \leq \sigma_S & \text{if } \operatorname{Dev}\dot{\boldsymbol{\varepsilon}} = \mathbf{0}. \end{cases} \tag{5.4}$$

This can be seen as the combination of a Newtonian viscosity with the inelastic associated flow rule deriving from the yield surface $F_{\text{Bi}}(\boldsymbol{\sigma}) := |\operatorname{Dev}\boldsymbol{\sigma}| - \sigma_S$.

The Bingham rheology (5.4) is a particular case of the Herschel-Buckley rheology, which is widely used for concrete or crude oil flows. Such industrial applications have made incompressible Bingham flows the subject of a very large body of research, and several numerical methods have been devised to handle to nonsmoothness of the rheology in an implicit manner; we refer to (Saramito and Wachs 2016) for a comprehensive review. Such methods can be classified into two main categories:

- Regularizing methods, which employ diverse numerical artifacts to smooth out the singularities of the rheology (see, e.g., (Frigaard and Nouar 2005) for a review). The main benefit of a regularizing strategy is the use of classical numerical schemes dedicated to differential equations. While some regularizing methods have been shown to converge to the true solution even in the ill-conditioned inviscid case (Bouchut, Eymard, et al. 2014), this limit cannot be realized numerically, and the regularization may predict yielding where none should occur (Frigaard and Nouar 2005).
- Nonsmooth methods, the first of which being Augmented Lagrangian methods (such as Arrow–Hurwitz or ADMM) based on the framework of (Fortin and Glowinski 1983), as used by Saramito and Roquet (2001). This class of methods is able to correctly predict the yielded and unyielded zones, but historically suffered from slow numerical convergence in practice. Recently however, several authors have focused on improving the convergence rate of non-regularizing approaches. Bleyer et al. (2015) proposed a method for Herschel–Bulkley fluids based on a SOCP reformulation, which they claim benefits from much faster convergence properties than Augmented Lagrangian methods. Saramito (2016) used a Newton algorithm on a complementarity function to solve for the steady-state of yield stress flows without Newtonian viscosity, and also observed improved convergence over the Augmented Lagrangian algorithm. Concurrently and in a similar fashion to our work, Treskatis et al. (2016) proposed the use of a Nesterov-accelerated proximal method on the dual minimization problem.

Granular materials Several works have focused on extending these numerical methods to the simulation of the non-associated Drucker–Prager or $\mu(I)$ [1] rheologies in 2D, inheriting in

[1] See Section 0.3.3.

the process the incompressibility constraint of Bingham flows (Chambon et al. 2011; Chauchat and Médale 2014; Ionescu et al. 2015; Lagrée et al. 2011; Staron et al. 2014). In Computer Graphics, Y. Zhu and Bridson (2005) used a similar approach for the simulation of 3D granular flows, enriching an incompressible flow solver with special treatment for zones below the yield stress.

However, incompressiblity of the flow means that the pressure field will only be defined up to a constant, and the usual zero-average condition will lead to the appearance of negative pressure values. This choice may prove unfortunate in certain configurations, such as in the wake of an obstacle, where special care has to be taken to ensure that the rheology remains well-defined (Chauchat and Médale 2014). Even then, using an incompressible flow model does not allow to capture the asymmetry of the pressure field experimentally observed by Seguin et al. (2016). Moreover, Barker et al. (2015) showed that the incompressible $\mu(I)$ rheology is instable for a wide range of parameters; they suggest that this instability may be alleviated by modeling the transitions between the different regimes of the granular material, and thus allowing dilation. In a very recent work, Heyman et al. (2016) also argue that relaxing the incompressibility constraint can resolve these stability issues.

In contrast, Narain, Golas, et al. (2010) proposed a flow-oriented approach that does not preclude the expansion of the material, and allows transitions between the solid, liquid and gaseous regimes. However, their approach was hampered by their use of a staggered time-integration procedure and a quite-rough approximation of the yield surface.

5.1.2 Elasto-plastic models

Another important body of research considers the plastic deformation of an elastic body. These approaches have been originally devised to study the behavior of metals outside of their elastic range, but can accommodate a much wider range of constitutive relationships, including that of granular materials (Vermeer 1998).

In contrast to flow-oriented approaches, these methods allow elastic displacement, and split the strain tensor into an elastic part $\boldsymbol{\varepsilon}_e$ and a plastic part $\boldsymbol{\varepsilon}_p$, using either additive or multiplicative plasticity theory. Using the terminology of the ISM framework, the bipotential b is usually chosen to be inviscid, and is defined from a *yield surface*, — i.e., a function F such that the set of admissible stresses is defined by $F(\boldsymbol{\sigma}) \leq 0$ — and a *flow rule*. Associated flow rules remain within the framework of GSM and are obtained by choosing the dissipation potential $\mathscr{D}$ such that $\mathscr{D}^\star = \mathscr{I}_{\{\boldsymbol{\tau}, F(\boldsymbol{\tau})\leq 0\}}$. In this case, Remark 1.2.1 states that Equation (5.3) can be written equivalently as (5.5),

$$\dot{\boldsymbol{\varepsilon}}_p \in \alpha \frac{\partial F}{\partial \boldsymbol{\sigma}}(\boldsymbol{\sigma}) \text{ and } 0 \geq F(\boldsymbol{\sigma}) \perp \alpha \geq 0. \tag{5.5}$$

Non-associated flow rules are also commonly used to model granular continua, and can be devised by replacing (5.3) with (5.6),

$$\dot{\boldsymbol{\varepsilon}}_p \in \alpha \frac{\partial G}{\partial \boldsymbol{\sigma}}(\boldsymbol{\sigma}) \text{ and } 0 \geq F(\boldsymbol{\sigma}) \perp \alpha \geq 0, \tag{5.6}$$

where the *plastic potential* G differs from F. The strains and stress at every point of the material are either computed explicitly — for instance by projecting the stresses onto the inside of the yield surface — or implicitly, using a so-called *return-mapping* algorithm (Simo and Hughes 2000), which usually takes the form of a root-finding Newton algorithm. Note that the non-associated Drucker–Prager flow rule, which we presented in the framework of ISM in Chapter 1, can also be written as (5.6). In this case, G is simply deduced from F by replacing the friction coefficient μ with the dilatancy coefficient ζ.

Mast (2013) compares different yield surfaces and flow rules (associated Drucker–Prager, Bingham or non-associated Natzuo–Makai), and proposes both implicit and explicit methods for their simulation. As for Drucker–Prager the plastic potential G is not differentiable at the "tip" of the cone of admissible stresses, $\frac{\partial G}{\partial \boldsymbol{\sigma}}$ may contain several elements, which complicates the return-mapping algorithm. Mast (2013, Section 3.6) proposes to use a two-surface approximation of

the yield-surface to alleviate this difficulty. Klar et al. (2016) argue that in the case of dry sand, the projection on the Drucker–Prager yield surface should be orthogonal to the hydrostatic axis $\sigma_1 = \sigma_2 = \sigma_3$. This corresponds indeed to an unassociated Drucker–Prager flow-rule with null dilatancy, as the projection is then analogous to that defining the Alart–Curnier complementarity function, as illustrated in Figure 1.3. Dunatunga and Kamrin (2015) explicitly models the transition between the gaseous and dense regimes based on the local volume fraction of grains, and restrict the use of the elasto-plastic flow rule to the latter regime.

This class of methods inherently model more complete physics, as it accounts for the measurable elasticity of the macroscopic material while flow-based methods do not. However, it suffers from the fact the compression Young modulus of granular materials made of rigid grains is very high, which leads to extremely stiff numerical systems; moreover, the time-scale of the elastic response is generally much lower than that of the plastic deformation, and not necessarily relevant for our applications. Using explicit integration schemes then requires taking very small timesteps, and implicit integration yields ill-conditioned, hard to solve systems. In practice, several works artificially lower the Young modulus for numerical efficiency purposes, weakening the physical correctness argument.

5.2 Spatial discretization strategies

While the above dichotomy focused on the constitutive law of the material (elastoplastic vs viscoplastic), we can also sort the different granular simulation methods following the strategy that they use to discretize the so-called *conservation* equations, which we first recall below.

5.2.1 Continuous conservation equations

The evolution of the material state in time is driven by two conservation equations. Let ρ denote the density of the material and $\boldsymbol{u}$ its velocity, then the conservation of mass reads

$$\frac{\partial \rho}{\partial t} + \nabla \cdot [\rho \mathbf{u}] = \mathbf{0}, \tag{5.7}$$

and the conservation of momentum (in conservative form) is

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot \left[\rho \mathbf{u} \otimes \mathbf{u} - \boldsymbol{\sigma}^{tot}\right] = \mathbf{f}^{ext}, \tag{5.8}$$

where $\boldsymbol{\sigma}^{tot}$ is the total stress tensor and $\mathbf{f}^{ext}$ the total density of external forces. These equations are derived from the application of basic physical principles over elementary volumes; we refer the interested reader to (Saramito 2013) for more details.

The momentum conservation equation (or *momentum balance*) (5.8) is typically written under another, *non-conservative* form, using the identity

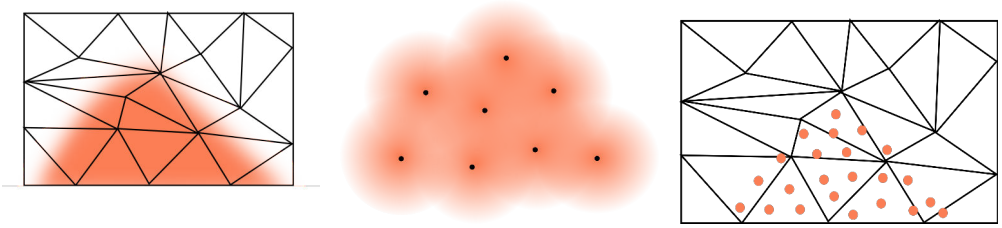
$$\begin{aligned} \nabla \cdot [\rho \mathbf{u} \otimes \mathbf{u}] &= \nabla \cdot [\rho \mathbf{u}]\, \mathbf{u} + (\rho \mathbf{u} \cdot \nabla)\, \mathbf{u} && \text{(see Remark 5.1)} \\ &= -\frac{\partial \rho}{\partial t} \mathbf{u} + \rho\, (\mathbf{u} \cdot \nabla)\, \mathbf{u} && \text{using (5.7).} \end{aligned} \tag{5.9}$$

Then using (5.9) inside the momentum balance (5.8), we get the non-conservative form,

$$\rho \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} \right) - \nabla \cdot \boldsymbol{\sigma}^{tot} = \mathbf{f}^{ext}. \tag{5.10}$$

Remark 5.1. *We use the classical notation* $(\mathbf{u} \cdot \nabla)\mathbf{v}$*, where* $\mathbf{u}$ *is a vector field (a "velocity") and* $\mathbf{v}$ *a scalar or vector field, to denote the quantity* $(\nabla \mathbf{v})\mathbf{u}$*, that is, the variation of* $\mathbf{v}$ *when traveling along the streamlines of the velocity field* $\mathbf{u}$*.*

To further simplify notation, we can also introduce the notion of total derivative.



(a) *Mesh-based* **(b)** *Particle-based* **(c)** *Hybrid mesh–particles*

Figure 5.1: *Different strategies for the discretization of the mass and momentum conservation equations.*

Definition 5.1. *Let* $\mathbf{u}$ *be a vector field and* $\mathbf{v}$ *a scalar or vector field, the total derivative of* $\mathbf{v}$ *w.r.t. time and the velocity field* $\mathbf{u}$ *will be denoted by*

$$\frac{D_{\mathbf{u}}\mathbf{v}}{D_{\mathbf{u}}t} := \frac{\partial \mathbf{v}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{v}.$$

When no confusion is possible, we will omit the velocity field and simply write $\frac{D\mathbf{v}}{Dt}$.

The total derivative expresses the total variation of the quantity $\mathbf{v}$ at a material point $\boldsymbol{x}(t)$, taking into account the fact that the material point moves with the velocity field $\mathbf{u}$, i.e., $\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = \mathbf{u}$. Using this new notation, the mass and momentum conservation equations can be written

$$\frac{D\rho}{Dt} + \rho\nabla\cdot\mathbf{u} = 0 \tag{5.11}$$

$$\rho\frac{D\mathbf{u}}{Dt} - \nabla\cdot\boldsymbol{\sigma}^{tot} = \mathbf{f}^{ext} \tag{5.12}$$

Now, there are several ways to discretize these equations in space and express the gradient and divergence operators. We first recall briefly the existence of two points of view for looking at the evolution of a dynamical system:

- With the *Lagrangian* point-of-view, which is classically used in DEM, the system is described w.r.t. the moving material points. As every material point knows its own history, the transport terms are easy to handle; however, the topology of the material may change over time, making the spatial differential operators difficult to evaluate.
- In contrast, the *Eulerian* point of view looks at the system from a fixed point of space. The material can be described w.r.t. a fixed domain, allowing for easy spatial differentiation, but making transport terms harder to treat.

In practice, many methods use combinations of these two points of view.

5.2.2 Mesh-based discretization

A first solution is, in an Eulerian fashion illustrated in Figure 5.1(a), to extrapolate the density, velocity and stress field over the whole simulation domain from their values at selected points on a mesh, the so-called *degrees of freedom*. The structure of the mesh can then be used to approximate the differential operators of the conservation equations using different strategies:

- finite-differences, where the differential operators are evaluated by computing the difference in the values at neighboring nodes of a regular grid, eventually staggered;
- finite-volumes, where the divergence operator is computed from the flux between neighboring mesh cells using Green's theorem (e.g., Lagrée et al. 2011);

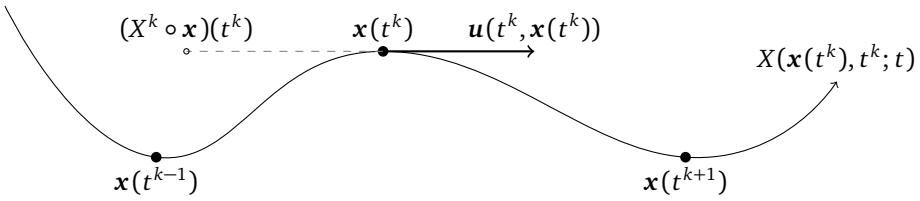


Figure 5.2: *Characteristics morphism X with its linearized version X^k*

- finite-elements (FEM) — of which the two previous items can actually be seen as special cases, where the conservation equations are solved under weak form, by looking at their image through their scalar product with a finite set of test functions (e.g., Ionescu et al. 2015).

Note that shallow models, which reduce a phenomenon whose depth is much smaller than its horizontal extent to a lower-dimensional problem, have also been developed with success for granular flows. Savage and Hutter (1989) were the first to propose one such model, assuming a proportional relationship of the components of the stress tensor — thus a flowing hypothesis. Ionescu (2010, 2013) studied the onset of the flow, with and without topography. Bouchut, Ionescu, et al. (2016) take a slightly different approach and consider a two-layer model, then study the depth of the interface between the solid and liquid regimes.

Transport terms The presence of total derivatives (transport terms) significantly complicates the numerical treatment of mesh-based conservation equations. The momentum balance equation is thus often studied in the *creeping flow* limit, when the inertial term $\rho(\boldsymbol{u}\cdot\nabla)\boldsymbol{u}$ is much smaller than the internal or external forces, but this strategy is not viable for all scenarios.

A convenient technique for the treatment of transport terms in timestepping schemes is the use of the *characteristics* method (see Etienne 2004), which brings insight from the Lagrangian point of view. The velocity and/or density fields are then advected following a backtracking strategy based on a low-order approximation of the trajectory of material points during the previous timestep. As in Figure 5.2, let us denote by $X(\boldsymbol{x}_0, t_0; t)$ the position at every instant of the material point that is at $\boldsymbol{x}_0$ at t_0; X is the solution of the Cauchy problem

$$\begin{cases} X(\boldsymbol{x}_0, t_0, t_0) = \boldsymbol{x}_0 \\ \dfrac{\partial X}{\partial t}(\boldsymbol{x}_0, t_0, t) = \mathbf{u}\left(X\left(\boldsymbol{x}_0, t_0, t\right), t\right). \end{cases} \tag{5.13}$$

Etienne (2004, Section 2.1) argues that for a regular-enough velocity field $\boldsymbol{u}$ and a small-enough time interval, the solution is unique, and thus the characteristics morphism X is well-defined. Now, composition by X allows to go back from the total derivative to the usual partial derivative. Indeed, let $\mathbf{v}$ be a scalar or vector field, then using the notation $(\mathbf{v}\circ X)(\boldsymbol{x}, t_0; t) := \mathbf{v}\left(X\left(\boldsymbol{x}, t_0; t\right), t\right)$,

$$\frac{\partial \mathbf{v}\circ X}{\partial t}(\boldsymbol{x}, t_0; t) = \left[\frac{\partial \mathbf{v}}{\partial t} + \frac{\partial \mathbf{v}}{\partial \boldsymbol{x}}\frac{\partial X}{\partial t}\right]\left(X\left(\boldsymbol{x}, t_0; t\right), t\right) = \frac{D\mathbf{v}}{Dt}\circ X. \tag{5.14}$$

This identity yield a simple first-order discretization of the total derivative. Consider a finite time-step $[t^l, t^{k+1} = t^k + \Delta_t]$, then

$$\frac{D\mathbf{v}}{Dt}(\boldsymbol{x}, t^{k+1}) = \frac{\mathbf{v}(\boldsymbol{x}, t^{k+1}) - (\mathbf{v}\circ X)\left(\boldsymbol{x}, t^{k+1}; t^k\right)}{\Delta_t} + O(\Delta_t).$$

Now, computing $X\left(\boldsymbol{x}, t^{k+1}; t^k\right)$ amounts to solving a Cauchy problem, which is unwieldy; in practice a low-order approximation of this quantity is used, usually using a Runge-Kutta method.

At the first order,

$$\frac{D\mathbf{v}}{Dt}(\boldsymbol{x}, t^{k+1}) = \frac{\mathbf{v}(\boldsymbol{x}, t^{k+1}) - \mathbf{v}\left(X^k(\boldsymbol{x}), t^k\right)}{\Delta_t} + O(\Delta_t), \quad X^k(\boldsymbol{x}) := \boldsymbol{x} - \Delta_t \mathbf{u}^k(\boldsymbol{x}, t^k). \tag{5.15}$$

Note that this requires being able to locate arbitrary points in the mesh, which can be costly and require using a dedicated spatial index structure. In a similar manner, one may evolve the mesh after each timestep, moving its nodes along the velocity field in a Lagrangian manner and thus yielding an explicit access to the characteristics morphism X. However, for flows incurring large displacements, as is often the case with granular materials, the mesh can quickly become degenerate and require remeshing.

Other approaches discretize directly the transport operators – possibly without resorting to timestepping scheme — in the Eulerian framework, but have to take special care to ensure energy and mass conservation yet prevent overshooting (the advected quantities should be physically admissible, for instance the density should remain positive). Saramito (2013, Section 4.10) illustrates that centered methods, such as finite-differences or continuous finite-elements, lead to parasitic oscillations on transport equations. Conversely, upwind discontinuous Galerkin methods (a finite-element method which can be interpreted as an arbitrary-degree generalization of finite-volumes, and which we will discuss briefly in Section 7.2.2) have been shown to possess good stability properties when discretizing transport terms (Pietro and Ern 2011). Specialized methods have also been devised for convection-dominated equations such as the mass conservation (5.7), for instance the WENO schemes (Shu 2009).

Note that when considering an incompressible flow with a spatially constant initial density, the mass conservation equation becomes trivial, as the density simply remains constant through time. This makes mesh-based approaches especially suited for simulating granular materials as incompressible yield-stress flows. However, if the portion of space occupied by the material is moving, one still has either to deform the mesh (e.g., Ionescu et al. 2015), or advect a boundary (e.g., Lagrée et al. 2011).

Visualization For Computer Graphics applications, the ultimate goal is to produce a set of grain samples that can be rendered. While it is easy to sample a density field at each individual frame, doing so in a temporally consistent manner is a hard problem; this drawback has also contributed to spark interest in particle-based approaches.

5.2.3 Particle-based discretization

Smoothed Particle Hydrodynamics (SPH) methods take a different approach, and remove the need for any structured mesh by making further use of the Lagrangian point of view. The material is discretized as a finite set of particles that represent clumps of material rather than individual grains. The velocity, stress and density fields are then discretized as the sum of smooth, compact, radially-symmetric kernels centered on those particles, as in Figure 5.1(b).

As such, the results are intrinsically free from grid artifacts. Moreover, this approach makes the transport terms trivial to discretize. Indeed, solving the mass conservation equations just amounts to moving the particles. Using an implicit Euler integration rule and denoting by $\boldsymbol{x}_p^k(t)$ and $\boldsymbol{v}_p^k(t)$ the position and velocity of the p^{th} particle at time t^k, this means

$$\boldsymbol{x}_p^{k+1} = \boldsymbol{x}_p^k + \Delta_t \boldsymbol{v}_p^k.$$

The particles also yield a direct access to the characteristics morphism (5.13), and the velocity's total derivative in the momentum balance equation can thus be simply discretized as

$$\frac{D\mathbf{u}}{Dt}(\boldsymbol{x}_p(t^{k+1}), t^{k+1}) = \frac{\boldsymbol{v}_p(t^{k+1}) - \boldsymbol{v}_p(t^k)}{\Delta_t} + O(\Delta_t).$$

Several authors have applied SPH to granular simulation (Alduán and Otaduy 2011; Alduán et al. 2009; Chambon et al. 2011; Ihmsen et al. 2013; Lenaerts and Dutré 2009). However, such methods typically require more degrees of freedom (particles) than mesh-based methods. Evaluating the differential operators require keeping track of each pair of neighboring particles, which can be costly. Enforcing constraints on the density is usually done with predictive-corrective approaches, requiring a lot of iterations with slow convergence.

5.2.4 Hybrid methods

Hybrid methods, illustrated on Figure 5.1(c), attempt to strike a compromise between the two aforementioned approaches. They still use particles for the representation of the material state — thus allowing easy handling of the transport terms, but augment them with a background mesh to help with the computation of differential operators and the internal forces. At each step, the new velocities are computed on the mesh, and then transferred to the particles.

The main difficulty of this approach thus lies in the particles-to-mesh and mesh-to-particles transfers; they should conserve energy, yet be stable. The first method of this kind, the Particle-in-Cell (PIC; Harlow 1963) method, used finite-differences on a regular grid for the mesh-based portion, and simple interpolation for transferring the velocities both ways.

We assume that the j^{th} component of a field $\mathbf{g}_j$ at any point in space can be evaluated by extrapolating its values $\mathbf{g}_{i,j}$ at the degrees of freedom $(\boldsymbol{y}_i)$ using the shape functions $\omega^v_{i,j}$, i.e., $\mathbf{g}_j(\boldsymbol{x}, t^k) = \sum_i \boldsymbol{g}^k_{i,j}\omega^v_{i,j}(\boldsymbol{x})$. A any timestep t^k, PIC transfers the velocities of the particles to the grid as a weighted average,

$$\boldsymbol{u}^{p\mapsto g,k}_{i,j} = \frac{\sum_p m_p \omega^v_{i,j}(\boldsymbol{x}^k_p)\boldsymbol{v}^k_{p,j}}{\sum_p m_p \omega^v_{i,j}(\boldsymbol{x}^k_p)}, \tag{5.16}$$

where m_p denotes the mass of the p^{th} particle.

The total velocity derivative can then be approximated to the first order as

$$\frac{D\mathbf{u}}{Dt}(\boldsymbol{x}, t^{k+1}) = \frac{\mathbf{u}(\boldsymbol{x}, t^{k+1}) - \mathbf{u}^{p\mapsto g}(\boldsymbol{x}, t^k)}{\Delta_t} + O(\Delta_t).$$

Finally, particles gets their velocities from the newly computed velocity field,

$$\boldsymbol{v}^{k+1}_p = \mathbf{u}(\boldsymbol{x}^k_p, t^{k+1}) \tag{5.17}$$

are moved using a semi-implicit Euler step,

$$\boldsymbol{x}^{k+1}_p = \boldsymbol{x}^k_p + \Delta_t \boldsymbol{v}^{k+1}_p. \tag{5.18}$$

However, this strategy was found to quickly dissipate kinetic energy. This led to the introduction of the FLuid-Implicit-Particle (FLIP) method (Brackbill and Ruppel 1986); instead of transferring the velocities from the grid to the particles, the particle velocities are updated using the difference in grid velocity from the previous to the current timestep. That is, Equation (5.17), is replaced with (5.19),

$$\boldsymbol{v}^{k+1}_p = \boldsymbol{v}^k_p + \left(\mathbf{u}\left(\boldsymbol{x}^k_p, t^{k+1}\right) - \mathbf{u}^{p\mapsto g}\left(\boldsymbol{x}^k_p, t^k\right)\right) \tag{5.19}$$

While this second velocity update rule leads to a much better conservation of kinetic energy, FLIP suffers from being prone to instabilities, losing the high-frequency filtering properties of PIC (Jiang, Schroeder, Selle, et al. 2015). In practice, a weighted average of the PIC and FLIP update rules is generally used. More recently, Jiang, Schroeder, Selle, et al. (2015) introduced the Affine Particle-in-Cell method (APIC). In this case, the grid-to-particles transfer use the original PIC rule (5.17), but an additional term, modeling the velocity gradients, is added to the particle-to-grid velocity transfers (5.16). This approach boasts a much better energy conservation than PIC, while preserving its stability and filtering properties.

The Material-Point Method (MPM; Sulsky 1994) and its derivatives, on which we will expand in Chapter 7, is another refinement of the PIC method that allows the usage of other mesh-based discretization strategies, such as the finite-element method. Any of the previously mentioned grid transfer strategies, PIC, FLIP and APIC, can also be used. The PIC particles-to-mesh transfer (5.16) can then be interpreted as using the *lumped* mass-matrix, corresponding to a trapezoidal approximation of the finite-element integral. As we will in Chapter 7, using an exact integration rule (*consistent* mass-matrix) with the PIC velocity update (5.17) preserves the kinetic energy, but yields a singular system when there are not enough particles in any given mesh cell. MPM has been often leveraged for the simulation of granular materials, mainly using elasto-plastic models (e.g., (S. Bardenhagen et al. 2000; Dunatunga and Kamrin 2015; Klar et al. 2016; Wieckowski, Youn, et al. 1999)) and explicit time-integration.

5.3 Our approach

We will dedicate the two following chapters to the devising of a simulation method for dry granular materials that meets the following design goals.

5.3.1 Design goals

Inelasticity Following the flow-oriented approaches, we will not attempt to capture elasticity of the material, but rather claim that for most applications the solid regime may be considered as purely rigid — which is a valid assumption for sand-like materials at low confining pressures (Roux and Radjai 1998). This will avoid us having to model the elasticity time scale, and thus allow the use of rougher time discretization schemes for increased computational performance. Moreover, we will not regularize the resulting multi-valued constitutive law. Instead, we claim that tackling it in an *implicit* manner by leveraging tools from nonsmooth optimization will yield better conditioned systems and stable results, even for large timesteps. Finally, similarly as for the discrete contact mechanics setting, we shall assume impacts to be inelastic, that is, shocks will not propagate inside the granular medium.

Dilatability and regime switching In contrast with most flow-oriented methods and following (Narain, Golas, et al. 2010), our method will not preclude the expansion of the flow, allowing the material to transition freely from a dense regime to a gaseous regime. This strategy avoids negative pressure zones and the ill-definition of the yield stress in these regions, and alleviates the instability of the incompressible $\mu(I)$ rheology.

A few existing approaches explicitly model abrupt transitions between the gaseous and dense (i.e., solid or liquid) regimes. They do so by considering a critical value for the density below which the grains are assumed to never be in persistent contact, and thus the contact stress vanishes. Switching between these two regimes may be done either in an explicit (Dunatunga and Kamrin 2015) or implicit (Narain, Golas, et al. 2010) manner; for numerical stability reasons we will follow the latter strategy. Moreover, in order to be consistent with our inelastic impact hypothesis, no rebound shall occur when the material suddenly enters the dense regime.

Computers graphics The different regimes exhibited by granular materials yield very rich dynamics, and their ubiquity in outdoor environments has made their visually plausible simulation a primary goal of the Computer Graphics community.

As we already mentioned, Computer Graphics is subject to different constraints and targets than engineering communities. While the physical accuracy criterion is less drastic in movies than risk-assessment applications, stability, robustness and numerical efficiency are prime requirements for visual effects pipelines. Obviously, any kind of visual artifacts is proscribed.

One of our design goals is to devise a method that is not only physically sound, but that should also be able to serve as a basis for Computer Graphics applications.

5.3.2 Outline of this second part

In Chapter 6, we will first consider dense granular materials, that is, materials that are already tightly packed together and cannot be compacted anymore. We will relax this assumption in Chapter 7, and consider the unconstrained flow of granular materials, but assume that the interactions with the surrounding fluid are negligible (which is the case for large-enough grains in the air). Finally, in Chapter 8 we will relax this second assumption, and model fully coupled flows such as powders in the air or immersed avalanches.

6 *Dense granular flows*

In this chapter, we consider a first simplified set of equations that is paradoxical, yet will allow us to retrieve classical results in a variety of scenarios, and will serve as a building block for more complex or accurate models. Most of the contents of this chapter have been published in (Daviet and Bertails-Descoubes 2016b).

6.1 Constitutive equations

6.1.1 Unilateral incompressibility

Let ϕ denote the volume fraction field, that is, the fraction of space occupied by the granular material at every point of the domain. ϕ can take values in $[0, \phi_{\max}]$, where $\phi_{\max} \leq 1$ is a maximum packing fraction depending on the geometry of the grains (for rigid spheres all of identical size, $\phi_{\max}$ cannot be above 0.74). Assuming a constant density ρ for the grains, the density field ρ can be computed as $\rho(\phi) = \rho\phi$, and thus the mass conservation equation (5.7) reads

$$\frac{D\phi}{Dt} = -\phi \nabla \cdot \mathbf{u}. \tag{6.1}$$

Dense flow hypothesis We consider that the material is, everywhere on the simulation domain and at every instant in time, densely packed; that is, $\phi = \phi_{\max}$.

However, we want to take into account the asymmetric yielding behavior of granulars by allowing the onset of expansion, while strictly preventing compaction. Taking the derivative w.r.t. time of the maximum compaction constraint $\phi \leq \phi_{\max}$ for an already densely packed material yields $\frac{D\phi}{Dt} \leq 0$, that is, using (6.1),

$$\nabla \cdot \mathbf{u} \geq 0. \tag{6.2}$$

We will call (6.2) the *unilateral incompressibility constraint*.

Actually, for the material to remain densely packed everywhere, mass conservation (6.1) dictates $\nabla \cdot \mathbf{u} = 0$. The paradoxical nature of our model comes from the fact that we only require positive divergence of the flow — not null divergence — yet still assume $\phi = \phi_{\max}$ at every instant. This means that mass conservation will not hold inside the strictly dilating zones. Fortunately, this inconsistency will have little effect on the predicted flow inside the dense regions. As validated in our Section 6.5, our simplified model still remains applicable to a variety of relevant scenarios.

Now, the main advantage of this dense flow hypothesis is that it avoids having to couple the momentum and mass conservation equations, side-stepping several technical difficulties and allowing for a much more concise exposition of our implicit numerical method, which is the main contribution of this chapter. Moreover, by carefully choosing the temporal and spatial discretization strategies for the constraint $\phi \leq \phi_{\max}$, we will show in Chapter 7 that the framework presented here can be easily adapted to accommodate arbitrary flows.

Complementarity condition For standard incompressible flows, the pressure field p can be seen as a Lagrange multiplier enforcing the null-divergence condition. Here, we will instead

set the pressure to enforce the unilateral compressibility constraint (6.2), i.e., formulating the Karesh-Kuhn-Tucker conditions for this inequality,

$$\begin{cases} \mathrm{p} \geq 0 & \text{and } \nabla \cdot \mathbf{u} = 0 \quad \text{or} \\ \mathrm{p} = 0 & \text{and } \nabla \cdot \mathbf{u} > 0, \end{cases}$$

or, using an equivalent complementarity notation,

$$0 \leq \mathrm{p} \perp \nabla \cdot \mathbf{u} \geq 0.$$

This means that the dilating zones will feature a vanishing pressure, while those that tend to remain densely packed may exhibit positive pressure. With our dense flow hypothesis, this follows the suggestion by Drew (1983) that the pressure should vanish when $\phi < \phi_{\max}$. Cohesion may also be modeled by allowing negative values for the pressure field; more generally, we will use the complementarity condition

$$0 \leq \mathrm{p} + c \perp \nabla \cdot \mathbf{u} \geq 0, \tag{6.3}$$

which states that the flow has to overcome a negative pressure of $-c$ before starting to dilate.

Our results (Section 6.5) will show that relaxing the common incompressibility assumption $\nabla \cdot \mathbf{u} = 0$ prevents the arising of an ill-defined rheology in some typical scenarios such as the flow in the wake of an obstacle (Chauchat and Médale 2014), and allows us to correctly retrieve a vanishing pressure field in this region.

6.1.2 Friction

Similarly as in Ionescu et al. (2015), we consider a rheology combining a viscosity η with a yield-stress $\kappa(\sqrt{\frac{d}{2}}p)$ that depends linearly on the local pressure p,

$$\kappa(x) := \sigma_S + \mu \left(x + \sqrt{\frac{d}{2}} c \right).$$

However, here we consider that η does not depend on the local velocity or pressure. We have chosen to incorporate the factor $\sqrt{\frac{d}{2}}$ in the above definitions in order for our physical parameters to match those of the non-associated Drucker–Prager rheology defined in Section 1.3, as we will see below. Similarly, the distinct roles of the two constant coefficients in κ, the cohesion c and the additional stress σ_S, will be made clear by using this analogy.

The total stress tensor $\boldsymbol{\sigma}^{\mathrm{tot}}$ is defined as [1]

$$\begin{aligned} \boldsymbol{\sigma}^{\mathrm{tot}} &:= 2\eta E^b(\boldsymbol{\varepsilon}) + \boldsymbol{\tau}^F - p\mathbb{I} \\ E^b(\boldsymbol{\varepsilon}) &:= \left(\mathrm{Dev} + \frac{b}{d}\mathbb{I}\,\mathrm{Tr} \right) \dot{\boldsymbol{\varepsilon}}, \qquad\qquad 0 < b \leq 1, \end{aligned}$$

where $\boldsymbol{\tau}^F$ is a traceless symmetric tensor bounded by the yield stress $\kappa(\sqrt{\frac{d}{2}}p)$ w.r.t. the norm $|\cdot|$ defined in Section (1.3.1),

$$|\boldsymbol{\tau}^F| = \sqrt{\langle \boldsymbol{\tau}^F, \boldsymbol{\tau}^F \rangle} = \sqrt{\frac{1}{2}\boldsymbol{\tau}^F : \boldsymbol{\tau}^F}.$$

Writing the maximum dissipation principle (i.e., the associated flow rule) for $\boldsymbol{\tau}^F$ yields

$$\begin{cases} \boldsymbol{\tau}^F = \kappa\left(\sqrt{\frac{d}{2}}\, p \right) \frac{\mathrm{Dev}\,\dot{\boldsymbol{\varepsilon}}}{|\mathrm{Dev}\,\dot{\boldsymbol{\varepsilon}}|} & \text{if } \mathrm{Dev}\,\dot{\boldsymbol{\varepsilon}} \neq \mathbf{0} \\ |\boldsymbol{\tau}^F| \leq \kappa\left(\sqrt{\frac{d}{2}}\, p \right) & \text{if } \mathrm{Dev}\,\dot{\boldsymbol{\varepsilon}} = \mathbf{0}. \end{cases} \tag{6.4}$$

[1] Using $E^b(\dot{\boldsymbol{\varepsilon}})$ rather than the usual $\mathrm{Dev}\,\dot{\boldsymbol{\varepsilon}}$ for the Newtonian viscosity models a non-zero bulk viscosity, and will ensure well-posedness of our equations (Propositon 6.2). However, in practice b can be chosen equal to 0.

Reformulation We will use once again the notions of *normal* and *tangential* parts of a symmetric tensor $\boldsymbol{\tau} \in S_d$ introduced in Section 1.3.1,

$$\boldsymbol{\tau}_{\mathrm{N}} := \langle \boldsymbol{\tau}, \boldsymbol{\iota}_d \rangle = \frac{1}{\sqrt{2d}} \operatorname{Tr} \boldsymbol{\tau} \qquad\qquad \boldsymbol{\tau}_{\mathbf{T}} := \operatorname{Dev} \boldsymbol{\tau}.$$

Using these notations, the total contact stress tensor, $\boldsymbol{\sigma}^c := \boldsymbol{\tau}^F - p\mathbb{I}$, satisfies $\boldsymbol{\sigma}^c_{\mathbf{T}} = \boldsymbol{\tau}^F$ and $\boldsymbol{\sigma}^c_{\mathrm{N}} = -\sqrt{\frac{d}{2}} p$. The frictional stress conditions (6.4) can be written as

$$\begin{cases} \boldsymbol{\sigma}^c_{\mathbf{T}} = \kappa\left(-\boldsymbol{\sigma}^c_{\mathrm{N}}\right) \dfrac{\dot{\boldsymbol{\varepsilon}}_{\mathbf{T}}}{|\dot{\boldsymbol{\varepsilon}}_{\mathbf{T}}|} & \text{if } \dot{\boldsymbol{\varepsilon}}_{\mathbf{T}} \neq \mathbf{0} \\ |\boldsymbol{\sigma}^c_{\mathbf{T}}| \leq \kappa\left(-\boldsymbol{\sigma}^c_{\mathrm{N}}\right) & \text{if } \dot{\boldsymbol{\varepsilon}}_{\mathbf{T}} = \mathbf{0}. \end{cases} \tag{6.5}$$

Moreover, since $\nabla \cdot \mathbf{u} = \operatorname{Tr} \dot{\boldsymbol{\varepsilon}} = \frac{1}{\sqrt{2d}} \dot{\boldsymbol{\varepsilon}}_{\mathrm{N}}$ and using the fact that complementarity is invariant to strictly positive scalings, Equation (6.3) can be written equivalently as

$$0 \leq \sqrt{\frac{d}{2}} c - \boldsymbol{\sigma}^c_{\mathrm{N}} \perp \dot{\boldsymbol{\varepsilon}}_{\mathrm{N}} \geq 0. \tag{6.6}$$

Putting (6.5) and (6.6) together, we can identify the disjunctive formulation (1.20) of the non-associated Drucker–Prager flow rule without dilatancy and with a tensile yield stress $\tau_c = \sqrt{\frac{d}{2}} c$. We will thus replace our rheology constraints (6.3–6.4) with the more general inclusion (6.7) allowing the modeling of a non-zero dilatancy ζ,

$$\left(\dot{\boldsymbol{\varepsilon}}, p\mathbb{I} - \boldsymbol{\tau}^F\right) \in \mathscr{DP}(\mu, \sigma_S, \tau_c, \zeta).$$

Finally, since we mentioned earlier that τ_c amounts to a simple translation of the set of admissible stresses along the hydrostatic axis, we will introduce the change of variable $\boldsymbol{\lambda} := \tau_c \boldsymbol{\iota}_d - \boldsymbol{\sigma}^c := (p + c)\mathbb{I} - \boldsymbol{\tau}^F$ and write our rheology as

$$(\dot{\boldsymbol{\varepsilon}}, \boldsymbol{\lambda}) \in \mathscr{DP}(\mu, \sigma_S, \zeta). \tag{6.7}$$

Cases covered by our choice of rheology Much like in (Ionescu et al. 2015), our set of parameters allows us to explore an interesting range of constitutive laws. When $\mu = 0$, we retrieve the viscoplastic Bingham rheology (with unilateral incompressibility – fully incompressible Bingham could be obtained in the $\tau_c = +\infty$ limit), while when taking $\eta = 0, \tau_c = 0$ and $\sigma_S = 0$, we get a purely Coulombic plastic flow — or, in other terms, the $\mu(I)$ rheology with identical static and dynamic friction coefficients. The numerical method presented in Section 6.2 will assume a non-zero η to obtain a well-posed system. However, our so-called "primal" algorithm will be able to handle a vanishing Newtonian viscosity, although some theoretical results will be lost. This constraint will be also be altogether alleviated in Section 6.5.3 when using temporal schemes. In our numerical experiments, we were also able to simulate the complete $\mu(I)$ rheology by explicitly computing the inertial number I at each time step. As the range of values for the $\mu(I)$ coefficient is relatively small, treating this term in an explicit fashion did not significantly degrade the stability of our simulations.

τ_c models cohesion (actually, a tensile yield strength), but the role of σ_S is different. For instance, a non-zero σ_S means that there can be a non-zero frictional stress for dilating flows. In contrast to τ_c, σ_S will induce a purely deviatoric stress. Moreover, as we saw in chapter 1, while τ_c amounts to a simple translation of the set of admissible stresses (which in our case will mean adding a confining pressure), σ_S fundamentally changes the structure of this admissible set, which becomes a truncated cone. σ_S is probably more relevant for concrete-like materials than granular ones; however, this parameter can also be useful in the 2D case to model wall friction or effects related to the geometry of the grains, which may induce a yield stress independent of the pressure in the simulation plane.

6.2 Creeping flow

In this first section we assume that the flow is slow enough for its inertia to be neglected, and solve for its steady state. While inertial effects are generally significant for granular flows, this case remains relevant since the structure of the equations for each timestep of the fully dynamic case will be similar (see Section 6.5.3).

6.2.1 Steady-state and boundary conditions

Let us consider a domain $\Omega \subset \mathbb{R}^d$ and decompose its boundary as $\mathrm{Bd}\,\Omega := B_D \cup B_N$, with Dirichlet boundary conditions on B_D and homogeneous Neumann on B_N,

$$\mathbf{u} = \mathbf{u}_D \qquad \text{on } B_D \tag{6.8a}$$

$$E^b\,\mathrm{D}(\mathbf{u})\,\boldsymbol{n}_\Omega = \mathbf{0} \qquad \text{on } B_N \tag{6.8b}$$

where $\boldsymbol{n}_\Omega$ is the outward-pointing normal to Ω on each point of its boundary.

Moreover, let $\boldsymbol{\sigma}^{\mathrm{ext}}$ gather all external stresses applied onto the Neumann boundary of the domain. We thus have

$$-\boldsymbol{\lambda}\,\boldsymbol{n}_\Omega = (\boldsymbol{\sigma}^{\mathrm{ext}} - c\mathbb{I})\,\boldsymbol{n}_\Omega \qquad \text{on } B_N. \tag{6.9}$$

We assume that the sole external force is the action of gravity whose orientation is given by the "down" unit vector $\boldsymbol{e}_g$. Writing the conservation of momentum (5.12) for the steady-state gives

$$-\nabla\cdot\Bigg[\underbrace{2\eta E^b\,\dot{\boldsymbol{\varepsilon}} - \boldsymbol{\lambda}}_{=\boldsymbol{\sigma}^{\mathrm{tot}} - c\mathbb{I}}\Bigg] = \rho g \boldsymbol{e}_g + \underbrace{\nabla c}_{\mathbf{0}} \qquad \text{on } \Omega. \tag{6.10}$$

From Equations (6.9) and (6.10), we see that the cohesion term c simply modifies the external stress term $\boldsymbol{\sigma}^{\mathrm{ext}}$; it the following, we will gather both of those terms into $\boldsymbol{\sigma}^{\mathrm{ext}}$ and ignore c.

Dimensionless equations Let L be a characteristic dimension of the flow. We define $U := \sqrt{gL}$ as the characteristic velocity and $P := \rho g L$ the characteristic pressure of the flow. We consider dimensionless differential operators defined through $\tilde{\nabla} := L\,\nabla$. We furthermore introduce two dimensionless numbers, the Reynolds number $\mathrm{Re} := \frac{\rho U L}{\eta}$ and the Bingham number $\mathrm{Bi} := \frac{\sigma_s}{\rho g L} = \frac{\sigma_s}{P}$.

Considering the dimensionless quantities $\tilde{\mathbf{u}} := \frac{1}{U}\mathbf{u}$, $\tilde{\dot{\boldsymbol{\varepsilon}}} := \tilde{\mathrm{D}}(\tilde{\mathbf{u}}) = \frac{L}{U}\dot{\boldsymbol{\varepsilon}}$, $\tilde{\boldsymbol{\lambda}} := \frac{1}{P}\boldsymbol{\lambda}$, and $\tilde{\mathbf{u}}_D := \frac{1}{U}\mathbf{u}_D$, Equations (6.8 – 6.10) can be made dimensionless as

$$\begin{cases} -\tilde{\nabla}\cdot\left[\dfrac{2}{\mathrm{Re}}E^b\,\tilde{\dot{\boldsymbol{\varepsilon}}} - \tilde{\boldsymbol{\lambda}}\right] = \boldsymbol{e}_g & \text{on } \Omega \\ \tilde{\mathbf{u}} = \tilde{\mathbf{u}}_D & \text{on } B_D \\ E^b\tilde{\mathrm{D}}(\tilde{\mathbf{u}})\,\boldsymbol{n}_\Omega = \mathbf{0} & \text{on } B_N \\ -\tilde{\boldsymbol{\lambda}}\,\boldsymbol{n}_\Omega = \tilde{\boldsymbol{\sigma}}^{\mathrm{ext}}\,\boldsymbol{n}_\Omega & \text{on } B_N, \end{cases} \tag{6.11}$$

with the dimensionless rheology

$$(\tilde{\dot{\boldsymbol{\varepsilon}}}, \tilde{\boldsymbol{\lambda}}) \in \mathscr{DP}(\mu, \mathrm{Bi}, \zeta). \tag{6.12}$$

From now on we shall use the dimensionless quantities and omit the tildes.

6.2.2 Variational formulation

Let $H^1(\Omega)^d$ be the usual Sobolev space containing square-integrable functions from $\Omega \subset \mathbb{R}^d$ to $\mathbb{R}^d$, with square-integrable gradients. As in (Saramito 2015, Appendix A), we introduce the affine subspace $V(\boldsymbol{u}_D)$ of $H^1(\Omega)^d$ for which the Dirichlet boundary condition (6.8a) is satisfied, i.e.,

$$V(\mathbf{u}_D) := \{\mathbf{u} \in H^1(\Omega)^d;\ \mathbf{u} = \mathbf{u}_D \text{ on } B_D\}.$$

$V(\mathbf{0})$ is therefore the vector subspace of $H^1(\Omega)^d$ for which the homogeneous Dirichlet boundary condition $\mathbf{u}_{|B_D} = \mathbf{0}$ is satisfied. Let also $T(\Omega)$ be the space of square-integrable symmetric tensor fields on Ω.

Proposition 6.1. *A weak form of System (6.11 – 6.12) amounts to finding* $\mathbf{u} \in V(\mathbf{u}_D)$, $\boldsymbol{\lambda} \in T(\Omega)$, *and* $\boldsymbol{\gamma} \in T(\Omega)$, *such that*

$$\begin{cases} a(\mathbf{u},\mathbf{v}) = b(\boldsymbol{\lambda},\mathbf{v}) + l(\mathbf{v}) & \forall \mathbf{v} \in V(\mathbf{0}) \quad (6.13a) \\ m(\boldsymbol{\gamma},\boldsymbol{\tau}) = b(\boldsymbol{\tau},\mathbf{u}) & \forall \boldsymbol{\tau} \in T(\Omega) \quad (6.13b) \\ (\boldsymbol{\gamma},\boldsymbol{\lambda}) \in \mathscr{DP}(\mu, Bi, \zeta), & (6.13c) \end{cases}$$

where $\forall \mathbf{x},\mathbf{y} \in H^1(\Omega)^d$ *and* $\forall \boldsymbol{\sigma},\boldsymbol{\tau} \in T(\Omega)$, $a(\boldsymbol{x},\boldsymbol{y})$ *and* $m(\boldsymbol{\sigma},\boldsymbol{\tau})$ *are the symmetric positive-definite bilinear forms on* $H^1(\Omega)^d \times H^1(\Omega)^d$ *and* $T(\Omega) \times T(\Omega)$, *respectively,*

$$a(\mathbf{x},\mathbf{y}) := \frac{2}{Re}\int_\Omega \mathrm{Dev}\,\mathrm{D}(\mathbf{x}) : \mathrm{Dev}\,\mathrm{D}(\mathbf{y}) + \frac{b}{d}(\nabla\cdot\mathbf{x})(\nabla\cdot\mathbf{y})$$
$$m(\boldsymbol{\sigma},\boldsymbol{\tau}) = \int_\Omega \boldsymbol{\sigma} : \boldsymbol{\tau},$$

$b(\boldsymbol{\tau},\mathbf{x})$ *is the bilinear form on* $T(\Omega) \times H^1(\Omega)^d$,

$$b(\boldsymbol{\tau},\mathbf{x}) = \int_\Omega \mathrm{D}(\mathbf{x}) : \boldsymbol{\tau},$$

and $l(\mathbf{x})$ *is the linear form on* $H^1(\Omega)^d$,

$$l(\mathbf{x}) = \int_\Omega \langle \boldsymbol{e}_g, \mathbf{x}\rangle - \int_{B_N} \langle (\boldsymbol{\sigma}^{ext}\boldsymbol{n}_\Omega), \boldsymbol{x}\rangle.$$

Proof. First, let us consider the stress boundary condition (6.9). One solution would be to enforce it strongly, by constraining $\boldsymbol{\lambda}$ to the subspace of $T(\Omega)$ which satisfy (6.9). However, this may lead to difficulties in the discretization of the $\mathscr{DP}(\mu, \mathrm{Bi}, \zeta)$ rheology. In our proposed implementation, we choose instead to model, in the integration of the term $\nabla\cdot\boldsymbol{\sigma}^{\mathrm{tot}}$ over Ω, a possibly non-zero jump $[\![\boldsymbol{\sigma}^{\mathrm{tot}}]\!] = -\boldsymbol{\lambda}_{|B_N} - \boldsymbol{\sigma}^{\mathrm{ext}}$ of the stress on B_N.

Let us now derive the variational formulation for System (6.11). We assume $\mathbf{u} \in V(\boldsymbol{u}_D)$ and $\boldsymbol{\lambda} \in T(\Omega)$, and let $\mathbf{v} \in V(\mathbf{0})$ be a test function. Multiplying both sides of the first line of (6.11) by $\boldsymbol{v}$ and integrating over Ω yields

$$-\int_\Omega \left\langle \nabla\cdot\left[\frac{2}{\mathrm{Re}}E^b\,\mathrm{D}(\mathbf{u}) - \boldsymbol{\lambda}\right], \boldsymbol{v}\right\rangle + \int_{B_N}\langle [\![\boldsymbol{\sigma}^{\mathrm{tot}}]\!]\boldsymbol{n}_\Omega, \mathbf{v}\rangle = \int_\Omega \langle \boldsymbol{e}_g, \mathbf{v}\rangle. \quad (6.14)$$

Using the Green formula with the Neumann (6.8b) and homogeneous Dirichlet (6.8a) boundary conditions for $\mathbf{u}$,

$$
\begin{aligned}
a(\mathbf{u},\mathbf{v}) &= \frac{2}{\mathrm{Re}}\int_\Omega \mathrm{Dev}\,\mathrm{D}(\mathbf{u}) : \mathrm{Dev}\,\mathrm{D}(\mathbf{v}) + \frac{b}{d}(\nabla\cdot\mathbf{v})(\nabla\cdot\mathbf{u}) \\
&= \int_\Omega \frac{2}{\mathrm{Re}} E^b\,\mathrm{D}(\mathbf{u}) : \mathrm{D}(\mathbf{v}) \\
&= -\int_\Omega \left\langle \nabla\cdot\left[\frac{2}{\mathrm{Re}}E^b\,\mathrm{D}(\mathbf{u})\right], \mathbf{v}\right\rangle
\end{aligned} \tag{6.15}
$$

and

$$
\int_\Omega \langle \nabla\cdot\boldsymbol{\lambda}, \mathbf{v}\rangle + \int_{B_N} \left\langle [\![\boldsymbol{\sigma}^{\mathrm{tot}}]\!]\,\boldsymbol{n}_\Omega, \mathbf{v}\right\rangle = -\int_{B_N} \left\langle \boldsymbol{\sigma}^{\mathrm{ext}}\boldsymbol{n}_\Omega, \mathbf{v}\right\rangle - b(\boldsymbol{\lambda},\mathbf{v}). \tag{6.16}
$$

By combining (6.14 – 6.16) and the definition of l, we retrieve (6.13a). Let us now focus on the rheology $\mathscr{DP}(\mu,\mathrm{Bi},\zeta)$ given in (6.12), which contains inequalities that cannot be put directly under weak form. To circumvent this difficulty, we introduce an auxiliary variable $\boldsymbol{\gamma}\in T(\Omega)$ that weakly satisfies $\boldsymbol{\gamma}=\mathrm{D}(\mathbf{u})$, i.e.,

$$
\int_\Omega \mathrm{D}(\mathbf{u}) : \boldsymbol{\tau} = \int_\Omega \boldsymbol{\gamma} : \boldsymbol{\tau} \quad \forall \boldsymbol{\tau}\in T(\Omega), \tag{6.17}
$$

which is exactly equation (6.13b). We can thus express the rheology $\mathscr{DP}(\mu,\mathrm{Bi},\zeta)$ under the weak form as (6.13b —6.13c). □

Remark Note that we do not include any additional equation ensuring the well-posedness of our system, such as the zero-average pressure condition which is commonly used for Stokes flows. Indeed, for a given velocity field, the $\mathscr{DP}(\mu,\mathrm{Bi},\zeta)$ rheology imposes the value of $\boldsymbol{\lambda}$ in the yielded regions, serving as an intrinsic boundary condition for the stress field inside the rigid zones.

6.2.3 Cadoux algorithm

In this section, we adapt the algorithm of Cadoux (2009) to show that solutions to our variational problem (6.13a–6.13c) can be characterized as the fixed-point of a sequence of strictly convex and well-posed minimization problems.

Proposition 6.2. *Let $s\in L_2(\Omega)$, let us use $\mathscr{K}_{\frac{1}{\mu}}$ as a short-hand for $\{\boldsymbol{\tau}\in T(\Omega), \boldsymbol{\tau}(\boldsymbol{x})\in\mathscr{K}_{\frac{1}{\mu(\boldsymbol{x})}}$ a.e. on $\Omega\}$. Under the regularity condition ($\mathscr{H}(s)$),*

$$
\exists \mathbf{u}\in V(\mathbf{u}_D), \mathrm{D}(\mathbf{u}) + s\boldsymbol{\iota}_d \in \mathrm{int}\,\mathscr{K}_{\frac{1}{\mu}}, \tag{$\mathscr{H}(s)$}
$$

and assuming $\mathrm{Re}<+\infty$, the minimization problem

$$
q(s) := \min_{\mathbf{u}\in C(s)} J(\mathbf{u}) \tag{6.18}
$$

$$
\begin{aligned}
J(\mathbf{u}) &:= \frac{1}{2}a(\mathbf{u},\mathbf{u}) - l(\mathbf{u}) + Bi\; g(\mathrm{D}(\mathbf{u})) \\
g(\boldsymbol{\gamma}) &:= 2\int_\Omega |\boldsymbol{\gamma}_T| \\
C(s) &:= \{\mathbf{u}\in V(\mathbf{u}_D), \mathrm{D}(\mathbf{u}) + s\boldsymbol{\iota}_d \in \mathscr{K}_{\frac{1}{\mu}}\}
\end{aligned} \tag{6.19}
$$

admits a unique solution which satisfies

$$
\begin{cases}
a(\mathbf{u},\mathbf{v}) = b(\boldsymbol{\lambda},\mathbf{v}) + l(\mathbf{v}) & \forall \mathbf{v}\in V(\mathbf{0}) \\
m(\boldsymbol{\gamma},\boldsymbol{\tau}) = b(\boldsymbol{\tau},\mathbf{u}) & \forall \boldsymbol{\tau}\in T(\Omega) \\
\boldsymbol{\gamma} + s\boldsymbol{\iota}_d \in -\mathscr{N}_{\mathscr{J}_{\mu,Bi}}(\boldsymbol{\lambda}).
\end{cases} \tag{6.20}
$$

Remark 6.1. *We readily deduce from Property 1.6 and Equation (6.20) that a solution* $\mathbf{u}(\mathrm{s})$ *to the minimization problem (6.18) such that* $\mathrm{s} = (\mu-\zeta)|\boldsymbol{\gamma}_T|$ *will yield a solution to our original variational problem (6.13a–6.13c). We can therefore define an operator* $F : L_2(\Omega) \to L_2(\Omega)$, $\mathrm{s} \mapsto (\mu-\zeta)|\,\mathrm{D}(u(\mathrm{s}))_T|$, *with* $u : \mathrm{s} \mapsto \arg\min_{\mathbf{u}\in C(\mathrm{s})} J(\mathbf{u})$, *so that any fixed-point of* F *will yield a solution to (6.13a–6.13c). Note that when* $\mu = \zeta$, *the fixed-point will be reached in a single step.*

Proof. As from Korn's inequality J is coercive and strictly convex, and under ($\mathscr{H}$(s)) the feasible set is not empty, Theorem (A.5) ensures that there exists a unique solution $\mathbf{u}$ to (6.18), which using Theorem A.6 can be characterized by

$$\mathbf{0} \in \partial J(\mathbf{u}) + \mathcal{N}_{C(\mathrm{s})}(\mathbf{u}). \tag{6.21}$$

$\mathrm{D}(\cdot) : H^1(\Omega)^d \to T(\Omega)$ and $\cdot_{|B_D} : H^1(\Omega)^d \to L^2(B_D)^d$ are linear operators from and to Hilbert spaces (trace theorem), we can thus use Corollary A.4 to Property A.12 on the normal cone to a precomposition by an affine map. Under the regularity condition ($\mathscr{H}$(s)),

$$\mathcal{N}_{C(\mathrm{s})}(\mathbf{u}) = \mathrm{D}(\cdot)^\intercal \mathcal{N}_{\mathscr{K}_{\frac{1}{\mu}}}(\mathrm{D}(\mathbf{u}) + \mathrm{s}\boldsymbol{\iota}_d) + (\cdot_{|B_D})^\intercal \mathcal{N}_{\{\mathbf{0}\}}(\mathbf{u}_{|B_D} - \mathbf{u}_D) \tag{6.22}$$

where $\mathrm{D}(\cdot)^\intercal$ denotes the adjoint operator to $\mathrm{D}(\cdot)$ for the usual scalar products of $H^1(\Omega)^d$ and $T(\Omega)$. Using once again Property A.12, we have also

$$\partial J(\mathbf{u}) = a(\mathbf{u},\cdot) - l + \mathrm{Bi}\,\mathrm{D}(\cdot)^\intercal \partial g(\mathrm{D}(\mathbf{u})), \tag{6.23}$$

where ∂g is also defined w.r.t. the usual scalar product of $T(\Omega)$, $\int_\Omega(\cdot:\cdot)$. Replacing (6.23) and (6.22) into (6.21), taking the scalar product with a test function $\mathbf{v} \in V(\mathbf{0})$ and noticing that $V(\mathbf{0}) = \mathrm{Ker}(\cdot_{|B_D}) = \left(\mathrm{Im}\,(\cdot_{|B_D})^\intercal\right)^\perp$, we get

$$(6.21) \iff \begin{cases} a(\mathbf{u},\mathbf{v}) = l(\mathbf{v}) + \left\langle \mathrm{D}(\cdot)^\intercal \boldsymbol{\lambda}, \mathbf{v}\right\rangle_{H^1} & \forall \mathbf{v} \in V(\mathbf{0}) \\ \boldsymbol{\gamma} = \mathrm{D}(\mathbf{u}) \\ \boldsymbol{\lambda} \in -\mathrm{Bi}\partial g(\boldsymbol{\gamma}) - \mathcal{N}_{\mathscr{K}_{\frac{1}{\mu}}}(\boldsymbol{\gamma} + \mathrm{s}\boldsymbol{\iota}_d). \end{cases} \tag{6.24}$$

Let us reformulate the last inclusion. We recognize from Section 1.3 and Equation (1.25) that for $\boldsymbol{\tau} \in S_d$,

$$\mathrm{Bi}(|\boldsymbol{\tau}_\mathbf{T}|) + \mathscr{I}_{\mathscr{K}_{\frac{1}{\mu}}}(\boldsymbol{\tau}) = \mathscr{I}_{-\mathscr{T}_{\mu,\mathrm{Bi}}}{}^\star(\boldsymbol{\tau}). \tag{6.25}$$

Using once again the notation $\mathscr{T}_{\mu,\mathrm{Bi}}$ as a shorthand for $\{\boldsymbol{\tau} \in T(\Omega), \boldsymbol{\tau}(\boldsymbol{x}) \in \mathscr{T}_{\mu(\boldsymbol{x}),\mathrm{Bi}}\ a.e.\}$, we get from Property A.5 on subdifferentials in function spaces that

$$\begin{aligned}
-\mathcal{N}_{\mathscr{T}_{\mu,\mathrm{Bi}}}(-\boldsymbol{\sigma}) &= \left\{\boldsymbol{\tau} \in T(\Omega), \boldsymbol{\tau}(\boldsymbol{x}) \in \mathcal{N}_{-\mathscr{T}_{\mu(\boldsymbol{x}),\mathrm{Bi}}}(\boldsymbol{\sigma}(\boldsymbol{x}))\ a.e.\ \text{on}\ \Omega\right\} \\
&= \left\{\boldsymbol{\tau} \in T(\Omega), \boldsymbol{\sigma}(\boldsymbol{x}) \in \partial \mathscr{I}_{-\mathscr{T}_{\mu,\mathrm{Bi}}}{}^\star(\boldsymbol{\tau}(\boldsymbol{x}))\ a.e.\right\} && \text{(Theorem A.2)} \\
&= \left\{\boldsymbol{\tau} \in T(\Omega), \boldsymbol{\sigma}(\boldsymbol{x}) \in \partial\left(\mathrm{Bi}(|\cdot_\mathbf{T}|) + \mathscr{I}_{\mathscr{K}_{\frac{1}{\mu}}}\right)(-\boldsymbol{\tau}(\boldsymbol{x}))\ a.e.\right\} && \text{using (6.25)} \\
&= \left\{\boldsymbol{\tau} \in T(\Omega), \boldsymbol{\sigma}(\boldsymbol{x}) \in \left(\mathrm{Bi}\partial(|\cdot_\mathbf{T}|) + \mathcal{N}_{\mathscr{K}_{\frac{1}{\mu}}}\right)(\boldsymbol{\tau}(\boldsymbol{x}))\ a.e.\right\} \\
&= \left\{\boldsymbol{\tau} \in T(\Omega), \boldsymbol{\sigma} \in \mathrm{Bi}\partial g(\boldsymbol{\tau}) + \mathcal{N}_{\mathscr{K}_{\frac{1}{\mu}}}(\boldsymbol{\tau})\right\}.
\end{aligned}$$

Note that the "2" in the definition of g is indeed retrieved in the last equality, as the scalar product and subdifferential on $T(\Omega)$ are defined w.r.t. the "$\cdot:\cdot$" scalar product on S_d, while the subdifferential of $|\cdot|$ on S_d is defined using the $\langle\cdot,\cdot\rangle = \frac{1}{2}\cdot:\cdot$ scalar product.

Hence, noticing that $\boldsymbol{\gamma}_\mathbf{T} = (\boldsymbol{\gamma} + \mathrm{s}\boldsymbol{\iota}_d)_\mathbf{T}$, we get the equivalence

$$-\boldsymbol{\lambda} \in \mathrm{Bi}\partial\left(\int |\cdot_\mathbf{T}|\right)(\boldsymbol{\gamma}) + \mathcal{N}_{\mathscr{K}_{\frac{1}{\mu}}}(\boldsymbol{\gamma} + \mathrm{s}\boldsymbol{\iota}_d) \iff \boldsymbol{\gamma} + \mathrm{s}\boldsymbol{\iota}_d \in -\mathcal{N}_{\mathscr{T}_{\mu,\mathrm{Bi}}}(\boldsymbol{\lambda}). \tag{6.26}$$

Finally, using that $\left\langle \mathrm{D}(\cdot)^{\intercal}\boldsymbol{\lambda}, \mathbf{v}\right\rangle_{H^1} = \langle \mathrm{D}(\mathbf{v}), \boldsymbol{\lambda}\rangle_{L_2} = b(\boldsymbol{\lambda}, \mathbf{v})$ and (6.26) to rewrite the first and last line of (6.24), respectively, we get that

$$(6.21) \iff \begin{cases} a(\mathbf{u}, \mathbf{v}) = b(\boldsymbol{\lambda}, \mathbf{v}) + l(\mathbf{v}) & \forall \mathbf{v} \in V(\mathbf{0}) \\ m(\boldsymbol{\gamma}, \boldsymbol{\tau}) = b(\boldsymbol{\tau}, \mathbf{u}) & \forall \boldsymbol{\tau} \in T(\Omega) \\ \boldsymbol{\gamma} + \mathfrak{s}\boldsymbol{\iota}_d \in -\mathscr{N}_{\mathscr{T}_{\mu,\mathrm{Bi}}}(\boldsymbol{\lambda}). \end{cases} \tag{6.27}$$

□

The F operator, as defined is Remark 6.1, allows us to define a Cadoux-like fixed-point algorithm.

Remark 6.2. *We used the coercivity of the bilinear form "a" to ensure the existence of a solution to each feasible optimization problem, and thus the well-posedness of the fixed-point algorithm. This condition is sufficient but not necessary, and in practice the method could be still be attempted for* $Re = +\infty$.

Existence of a solution to the variational problem ? In the discrete case, Cadoux (2009) proves the existence of a fixed-point for the F operator under the hypothesis $\mathscr{H}(\boldsymbol{s})$ by demonstrating the continuity and boundedness of F and applying the Brouwer fixed-point theorem. In our continuous case, one can also show relatively easily (at least for $\mathrm{Bi} = 0$) that the function $q : L_2(\Omega) \to \bar{\mathbb{R}}$ defined from the minimization problem (6.18) is continuous on $S_+ := \{s \in L_2(\Omega), s \geq 0\, a.e.\ \text{on}\ \Omega\}$, and that the application $u : S_+ \to H^1(\Omega)^d$, associating to $s \in S_+$ the unique solution to the minimization problem (6.18), is continuous and bounded on S_+. Since $F = f \circ u$, with $f : H^1(\Omega)^d \to S_+$, $\mathbf{u} \mapsto (\mu - \zeta)|\mathrm{D}(\mathbf{u})_{\mathbf{T}}|$, the existence of a solution to our flow problem under the hypothesis $\mathscr{H}(\mathrm{s})$ would be ensured by the existence of a fixed-point to $u \circ f$. This application looks indeed like a good candidate; $u \circ f$ is continuous and bounded on $H^1(\Omega)^d$, and the Rellich theorem states that its injection on $L_2(\Omega)^d$ is compact. However, the continuity of $u \circ f$ w.r.t. the L_2-norm does not seem obvious.

6.3 Discretization using finite-elements

In this section, we propose to discretize the variational problem (6.13a–6.13c) in such a way that the characterization of the solution as a fixed-point of a sequence of convex minimization problems (Remark 6.1) will still hold in the discrete case.

Moreover, we will see that the discrete version of the variational problem will have the same structure as the Discrete Coulomb Friction Problem (DCFP) that we studied in Chapters 2–4, allowing us to leverage similar algorithms.

6.3.1 Discretization of the symmetric tensor fields

For the discretization of the space $T(\Omega)$, we shall make use of Lagrange FEM, which means that all symmetric tensor fields will be expressed as the extrapolation over Ω of their values at n degrees of freedom $(\boldsymbol{y}_i) \in \mathbb{R}^{nd}$.

Let Q_h be a subspace of $L_2(\Omega)$ with dimension $n \in \mathbb{N}$, and $\left(\omega_i^{\tau}\right)_{1 \leq i \leq n}$ a basis of Q_h such that $\omega_i^{\tau}(\boldsymbol{y}_j) = \delta_{i,j}$. Let $(\boldsymbol{S}_j)_{1 \leq j \leq \mathfrak{s}_d}$ be a basis of S_d, with $\mathfrak{s}_d := \dim S_d = \frac{1}{2}d(d-1)$. We may build a finite subspace $T_h \subset T(\Omega)$ from the basis $(\mathbf{T}_k)_{1 \leq k \leq \mathfrak{s}_d\, n}$ defined as

$$\mathbf{T}_{\mathfrak{s}_d(i-1)+j} := \omega_i^{\tau} \boldsymbol{S}_j \quad \text{for } 1 \leq i \leq n, 1 \leq j \leq \mathfrak{s}_d.$$

Then $\forall \boldsymbol{\tau}_h \in T_h$, $\exists \underline{\boldsymbol{\tau}} := (\tau_k) \in \mathbb{R}^{n\,\mathfrak{s}_d}$ such that

$$\boldsymbol{\tau}_h = \sum_{\substack{1 \leq i \leq n \\ 1 \leq j \leq \mathfrak{s}_d}} \tau_{\mathfrak{s}_d(i-1)+j} \mathbf{T}_{\mathfrak{s}_d(i-1)+j} = \sum_{1 \leq i \leq n} \boldsymbol{\tau}_h(\boldsymbol{y}_i) \omega_i^{\tau},$$

$$\boldsymbol{\tau}_h(\boldsymbol{y}_i) = \sum_{1 \leq j \leq \mathfrak{s}_d} \tau_{\mathfrak{s}_d(i-1)+j} \boldsymbol{S}_j \qquad \forall 1 \leq i \leq n.$$

Choice of basis for S_d We want to leverage the analogy between the Drucker–Prager flow rule, which is expressed on tensors in S_d, and the Coulomb frictional contact law, which is expressed on vectors in $\mathbb{R}^d$. As such, will now construct a basis of S_d such that the properties of the "normal" and "tangential" parts are preserved.

Definition 6.1. *Let us introduce the morphism χ,*

$$\begin{aligned}
\chi : \mathbb{R}^{\mathfrak{s}_d} &\to S_d \\
(a;b,c) &\mapsto \begin{pmatrix} b & c \\ c & -b \end{pmatrix} + a\,\mathbb{I} && \text{if } d=2 \\
(a;b,c,d,e,f) &\mapsto \begin{pmatrix} b-\frac{c}{\sqrt{3}} & d & e \\ d & -b-\frac{c}{\sqrt{3}} & f \\ e & f & \frac{2c}{\sqrt{3}} \end{pmatrix} + \frac{\sqrt{2}}{\sqrt{3}} a\,\mathbb{I} && \text{if } d=3.
\end{aligned}$$

The morphism χ satisfies the following two properties:

Property 6.1. *χ is an orthonormal isomorphism between the two Euclidean spaces $(\mathbb{R}^{\mathfrak{s}_d}; \cdot^{\intercal}\cdot)$ and $(S_d; \langle\cdot,\cdot\rangle)$. This means*

$$\forall (\boldsymbol{x},\boldsymbol{y}) \in \mathbb{R}^{\mathfrak{s}_d} \times \mathbb{R}^{\mathfrak{s}_d}\ \boldsymbol{x}^{\intercal}\boldsymbol{y} = \langle \chi(\boldsymbol{x}), \chi(\boldsymbol{y}) \rangle, \tag{6.28}$$

where $\boldsymbol{x}^{\intercal}\boldsymbol{y}$ is the usual scalar product on $\mathbb{R}^m, m \geq 1$ and $\langle \boldsymbol{\sigma}, \boldsymbol{\tau} \rangle = \frac{\boldsymbol{\sigma}:\boldsymbol{\tau}}{2}$ is our scalar product on S_d.

Property 6.2. *Let $(\cdot)_N$ and $(\cdot)_T$ designate the normal and tangential parts of vectors and symmetric tensors as introduced in Chapter 1;*

$$\begin{aligned}
&\boldsymbol{n} := (1,0,\dots,0) \in \mathbb{R}^{\mathfrak{s}_d} && \boldsymbol{x}_N := \boldsymbol{n}^{\intercal}\boldsymbol{x} && \boldsymbol{x}_{\boldsymbol{T}} := \boldsymbol{x} - \boldsymbol{x}_N \boldsymbol{n} && \forall \boldsymbol{x} \in \mathbb{R}^m \\
&\boldsymbol{\iota}_d := \sqrt{\frac{d}{2}}\mathbb{I} \in S_d && \boldsymbol{\tau}_N := \langle \boldsymbol{\iota}_d, \boldsymbol{\tau} \rangle && \boldsymbol{\tau}_{\boldsymbol{T}} := \boldsymbol{\tau} - \boldsymbol{\tau}_N \boldsymbol{\iota}_d = \operatorname{Dev}\boldsymbol{\tau} && \forall \boldsymbol{\tau} \in S_d.
\end{aligned}$$

Then $\forall \boldsymbol{x} \in \mathbb{R}^m$,

$$\begin{cases} [\chi(\boldsymbol{x})]_N = \boldsymbol{x}_N & \text{(6.29a)} \\ \left|[\chi(\boldsymbol{x})]_{\boldsymbol{T}}\right| = \|\boldsymbol{x}_{\boldsymbol{T}}\|. & \text{(6.29b)} \end{cases}$$

It follows immediately from (6.29a–6.29b) that $\boldsymbol{\sigma} \in \mathscr{T}^{(S_d)}_{\mu,\mathrm{Bi}} \iff \chi^{-1}(\boldsymbol{\sigma}) \in \mathscr{T}^{(\mathbb{R}^{\mathfrak{s}_d})}_{\mu,\mathrm{Bi}}$. As χ is orthogonal (Property 6.1), the inclusion in normal cones is also preserved. Finally, by linearity of χ,

$$\begin{aligned}
& \boldsymbol{\gamma} + (\mu-\zeta)|\boldsymbol{\gamma}_{\mathbf{T}}|\boldsymbol{\iota}_d \in -\mathscr{N}_{\mathscr{T}^{(S_d)}_{\mu,\mathrm{Bi}}}(\boldsymbol{\sigma}) \\
\iff \quad & \chi^1\left(\boldsymbol{\gamma} + (\mu-\zeta)|\boldsymbol{\gamma}_{\mathbf{T}}|\boldsymbol{\iota}_d\right) \in -\mathscr{N}_{\mathscr{T}^{(\mathbb{R}^{\mathfrak{s}_d})}_{\mu,\mathrm{Bi}}}(\chi^{-1}(\boldsymbol{\sigma})) \\
\iff \quad & \chi^{-1}(\boldsymbol{\gamma}) + (\mu-\zeta)\left\|\left[\chi^{-1}(\boldsymbol{\gamma})\right]_{\mathbf{T}}\right\|\boldsymbol{n} \in -\mathscr{N}_{\mathscr{T}^{(\mathbb{R}^{\mathfrak{s}_d})}_{\mu,\mathrm{Bi}}}(\chi^{-1}(\boldsymbol{\sigma})).
\end{aligned}$$

We will thus be able to write the Drucker–Prager flow rule indifferently on symmetric tensors or on their image in $\mathbb{R}^{\mathfrak{s}_d}$ by the orthonormal isomorphism χ^{-1}, and, with a slight abuse of notations, use the equivalence

$$(\boldsymbol{\gamma}, \boldsymbol{\lambda}) \in \mathscr{DP}(\mu, \mathrm{Bi}, \zeta) \iff \left(\chi^{-1}(\boldsymbol{\gamma}), \chi^{-1}(\boldsymbol{\lambda})\right) \in \mathscr{DP}(\mu, \mathrm{Bi}, \zeta). \tag{6.30}$$

Going back to the discretization of our symmetric tensor fields, the equivalence (6.30) motivates choosing $\boldsymbol{S}_j := \chi(\boldsymbol{e}_j)$ as the basis for S_d, where $(\boldsymbol{e}_j)_{1\leq j\leq \mathfrak{s}_d}$ is the canonical basis of $\mathbb{R}^{\mathfrak{s}_d}$. Our global basis for T_h becomes

$$\mathbf{T}_{\mathfrak{s}_d(i-1)+j} := \omega_i^{\tau}\chi(\boldsymbol{e}_j) \quad \text{for } 1 \leq i \leq n, 1 \leq j \leq \mathfrak{s}_d.$$

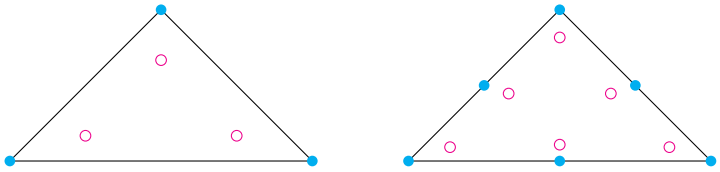


Figure 6.1: *Lagrange degrees of freedom (discs, blue) and Gauss quadrature points (circles, magenta) for two choices of triangular discretization. Left: piecewise-linear ($\mathbb{P}_1$) shape function and order-2 quadrature. Right: piecewise-quadratic ($\mathbb{P}_2$) shape function and order-4 quadrature.*

The Drucker–Prager rheology at the degrees of freedom $(\boldsymbol{y}_i)$ can now be directly expressed on the vectors of coefficients of the strain rate and stress fields. Indeed, denoting by $\underline{\boldsymbol{\tau}}_{[i]} \in \mathbb{R}^{\mathfrak{s}_d}$ the segment of $\mathfrak{s}_d$ coefficients of $\boldsymbol{\tau}_h$ corresponding to the i^{th} degree of freedom, $\underline{\boldsymbol{\tau}}_{[i]} := [\tau_{\mathfrak{s}_d(i-1)+j}]_{1\le j\le \mathfrak{s}_d}$, then $\forall 1 \le i \le n$, $\boldsymbol{\tau}_h(\boldsymbol{y}_i) = \chi(\underline{\boldsymbol{\tau}}_{\boldsymbol{h}_{[i]}})$, and thus using (6.30),

$$(\boldsymbol{\gamma}_h(\boldsymbol{y}_i), \boldsymbol{\lambda}_h(\boldsymbol{y}_i)) \in \mathscr{DP}(\mu, \mathrm{Bi}, \zeta) \iff (\underline{\boldsymbol{\gamma}}_{[i]}, \underline{\boldsymbol{\lambda}}_{[i]}) \in \mathscr{DP}(\mu, \mathrm{Bi}, \zeta). \tag{6.31}$$

6.3.2 Discretization of the (bi)linear forms

Let $U_h \subset H^1(\Omega)^d$ be a finite-dimensional vector space, and $V_h \subset U_h$ its subspace satisfying the Dirichlet boundary conditions

$$V_h(\mathbf{u}_{D,h}) := \left\{\mathbf{v}_h \in U_h, \mathbf{v} = \mathbf{u}_{D,h} \text{ on } B_D\right\},$$

where $\mathbf{u}_{D,h}$ discretizes $\mathbf{u}_D$ on the FEM mesh. Let $\mathbf{u}^*_{D,h} \in V_h(\mathbf{u}_{D,h}) \cap V_h(\mathbf{0})^\perp$ i.e., the unique function such that $V_h(\mathbf{u}_{D,h}) = V_h(\mathbf{0}) + \{\mathbf{u}^*_{D,h}\}$.

Unlike for the space T_h, here we make no specific assumption regarding the structure of $V_h(\mathbf{0})$ or the construction of a corresponding basis. Let $\mathfrak{v} := \dim V_h$. Given $(\mathbf{V}_i)_{1\le i\le \mathfrak{v}}$ a basis of $V_h(\mathbf{0})$, we denote by A, B, and M the matrices corresponding to the decomposition of the bilinear forms a, b, and m, respectively, and by $\underline{\boldsymbol{l}}$ the vector corresponding to the decomposition of the linear form l. More precisely, we have $A_{i,j} = a(\mathbf{V}_i, \mathbf{V}_j)$, $M_{k,\ell} = m(\mathbf{T}_k, \mathbf{T}_\ell)$, $B_{k,j} = b(\mathbf{T}_k, \mathbf{V}_j)$ and $\underline{\boldsymbol{l}}_j = l(\mathbf{V}_j)$. Similarly, let $\underline{\boldsymbol{u}}$ be the vector of scalar coefficients corresponding to the decomposition of the projection of $\mathbf{u}_h$ on the basis $(\mathbf{V}_i)$ of $V_h(\mathbf{0})$. We have $\mathbf{u} = \sum_{1\le i\le \mathfrak{v}} \underline{\boldsymbol{u}}_i \mathbf{V}_i(x) + \mathbf{u}^*_{D,h}$.

FEM discrete system At this point the discrete version of Equations (6.13a – 6.13b) reads: Find $\mathbf{u}_h \in V_h(\mathbf{0})$, $(\boldsymbol{\lambda}_h, \boldsymbol{\gamma}_h) \in T_h^2$,

$$\begin{cases} A\underline{\boldsymbol{u}} = B^\intercal \underline{\boldsymbol{\lambda}} + \underbrace{\underline{\boldsymbol{l}} - \underline{\boldsymbol{a}}_D}_{\underline{\boldsymbol{l}}^{\text{tot}}} & \text{(6.32a)} \\ M\underline{\boldsymbol{\gamma}} = B\underline{\boldsymbol{u}} + \underline{\boldsymbol{k}} & \text{(6.32b)} \\ (\boldsymbol{\gamma}_h, \boldsymbol{\lambda}_h) \in \text{discrete version of } \mathscr{DP}(\mu, \mathrm{Bi}, \zeta) & \text{(6.32c)} \end{cases}$$

where for $1 \le i \le \mathfrak{v}$, $\underline{\boldsymbol{a}}_{D,i} = a(\mathbf{u}^*_{D,h}, \mathbf{V}_i)$ and for $1 \le k \le \mathfrak{s}_d\, n$, $\underline{\boldsymbol{k}}_k = b(\mathbf{T}_k, \mathbf{u}^*_{D,h})$. It now remains to discretize the rheology constraint, $(\boldsymbol{\gamma}, \boldsymbol{\lambda}) \in \mathscr{DP}(\mu, \mathrm{Bi}, \zeta)$, that is, to wrtie an explicit expression for Equation (6.32c).

6.3.3 Discretization of the Drucker–Prager flow rule

As $\mathscr{DP}(\mu, \mathrm{Bi}, \zeta)$ is non-convex, the inclusion $(\boldsymbol{\gamma}_h, \boldsymbol{\lambda}_h) \in \mathscr{DP}(\mu, \mathrm{Bi}, \zeta)$, where the values of $\boldsymbol{\gamma}_h$ and $\boldsymbol{\lambda}_h$ are extrapolated between degrees of freedom, will not be able to hold at every point in space. Instead, we will attempt to satisfy the rheology in a weaker sense.

Remember from Chapter 1 that the Drucker–Prager flow rule can be expressed as a root-finding problem, for instance on the De Saxcé complementarity function (1.29),

$$(\boldsymbol{\gamma}, \boldsymbol{\lambda}) \in \mathscr{DP}(\mu, \mathrm{Bi}, \zeta) \iff f_{\mathrm{DS}}(\boldsymbol{\gamma}, \boldsymbol{\lambda}) = \mathbf{0}.$$

We can thus write a weak form of (6.32c) as

$$\int_\Omega f_{\mathrm{DS}}(\boldsymbol{\gamma}_h, \boldsymbol{\lambda}_h) : \boldsymbol{\tau}_h = 0 \quad \forall \boldsymbol{\tau}_h \in T_h\,,$$

which amounts to saying that the projection of $f_{\mathrm{DS}}(\boldsymbol{\gamma}_h, \boldsymbol{\lambda}_h)$ on T_h should vanish.

If f_{DS} were piecewise-polynomial, we could reduce these integrals to the evaluation of the integrand at a discrete number of quadrature points $(\check{\boldsymbol{x}}_q)$, and thus discretize the constraint as

$$f_{\mathrm{DS}}(\boldsymbol{\gamma}_h(\check{\boldsymbol{x}}_q), \boldsymbol{\lambda}_h(\check{\boldsymbol{x}}_q)) = 0 \quad \forall q. \tag{6.33}$$

Obviously, f_{DS} is not piecewise-polynomial, but we we shall still restrict ourselves to a discrete number of points $\check{\boldsymbol{x}}_q$ at which to enforce $(\boldsymbol{\gamma}_h(\check{\boldsymbol{x}}_q), \boldsymbol{\lambda}_h(\check{\boldsymbol{x}}_q)) \in \mathscr{DP}(\mu, \mathrm{Bi}, \zeta)$ — in a sense, saying that a "projection" of f_{DS} onto a polynomial space should be zero. The remaining question is what makes a good choice for this set of points $(\check{\boldsymbol{x}}_q)$. In the following we shall see that this choice has a direct impact on the final form of the numerical system that must be solved, and thus on both the physical relevance of the discrete problem and the computational performance of solving methods.

Discretization on Q_h's Lagrange degrees of freedom One obvious choice for $(\check{\boldsymbol{x}})_q$ is to consider the n degrees of freedom $(\boldsymbol{y}_i)$ which served to define the $(\omega^\tau)_i$ basis and thus the finite-dimensional spaces Q_h and T_h. Indeed, let $\boldsymbol{\tau}_h \in T_h$ such that $\boldsymbol{\tau}(\boldsymbol{y}_i) = \mathbf{0}$ $\forall 1 \le i \le n$, then $\boldsymbol{\tau}_h = \mathbf{0}$. From (6.31), replacing (6.32c) with

$$\left(\underline{\boldsymbol{\gamma}}_{[i]}, \underline{\boldsymbol{\lambda}}_{[i]}\right) \in \mathscr{DP}(\mu_i, \mathrm{Bi}_i, \zeta_i) \quad \text{for} \quad 1 \le i \le n.$$

thus means that any $\boldsymbol{\tau}_f \in T_h$ interpolating $f_{\mathrm{DS}}(\boldsymbol{\gamma}_h, \boldsymbol{\lambda}_h)$ at the degrees of freedom satisfies $\boldsymbol{\tau}_h = \mathbf{0}$.

With this choice of discretization for the constraint, the discrete system (6.32a–6.32c) looks a lot like a DCFP, with one notable difference: the presence of the M matrix. Since A and M are positive-definite, we may eliminate the velocity variable $\underline{\boldsymbol{u}}$ from (6.32a), and then get a linear relationship between $\underline{\boldsymbol{\lambda}}$ and $\underline{\boldsymbol{\gamma}}$ from (6.32b), $\underline{\boldsymbol{\gamma}} \propto W\underline{\boldsymbol{\lambda}}$ where $W = M^{-1}BA^{-1}B^\intercal$. Continuing the DCFP analogy, the "Delassus" operator W of (6.32a–6.32c) is generally not symmetric, as M and $BA^{-1}B^\intercal$ do not necessarily commute. This lack of symmetry is problematic; indeed, consider the case $\mu = \zeta$, the discrete problem reduces to

$$W\underline{\boldsymbol{\lambda}} + \underline{\boldsymbol{b}} \in -\mathscr{N}_{\mathscr{D}_{\mu,\mathrm{Bi}}}(\underline{\boldsymbol{\lambda}}),$$

with $\boldsymbol{b}$ a constant vector in $\mathbb{R}^{nd}$. As W is asymetric, this does not correspond to the optimality conditions of a convex minimization problem; more generally, the Cadoux algorithm that we devised in the continuous case can no longer be applied.

The discrete system (6.32) therefore lacks a fundamental symmetry property, which is key not only to guarantee physical consistency of our model, but also to the design of efficient numerical solvers. From a physical point of view, such an asymmetry in our discrete frictional contact law typically implies that the maximum dissipation principle cannot be satisfied, meaning that some anisotropy is artificially introduced through the discretization. From a purely numerical point of view, symmetry of the Delassus operator is not necessarily a prerequisite to common numerical solvers, but in our case it proves to be highly desirable for coming up with a tractable solving method. Indeed, among scalable solvers for DCFP, we basically have the choice between operator-splitting algorithms (Section (3.4)), and optimization-based algorithms (Sections 3.2 and 3.3). On the one hand, as mentioned above the asymmetric nature of

W disqualifies the interpretation of as the fixed-point of a sequence of minimization problems, and thus optimization-based methods have to be discarded. On the other hand, the Gauss–Seidel splitting algorithm does not require W to be symmetric (except for the interpretation as block coordinate-descent), yet for efficiency purposes it requires the explicit knowledge of W or a at least a very cheap way to compute matrix-vector products with $A^{-1}B^{\mathsf{T}}$. In our case A^{-1} is dense, and so is W, making the Gauss–Seidel algorithm intractable, especially for the large systems that we are going to have to deal with. For these reasons, we choose to discretize our constraints on an alternative set of points $(\check{\boldsymbol{x}})_q$ that will allow us to eliminate the matrix M and thus retrieve symmetry of W. This way we shall both recover physical consistency of our model, and benefit from efficient optimization-based solving methods.

Discretization on Gauss quadrature points As we are using a Lagrange FEM discretization of the space $T(\Omega)$ with polynomial interpolating bases, each integral $M_{k,\ell} = \int \mathbf{T}_k : \mathbf{T}_\ell$ can be computed exactly using Gaussian quadrature of order $2r$, where r is the degree of the piecewise polynomials in T_h. That is, $M_{k,\ell} = \sum_q w_q \left(\mathbf{T}_k(\hat{\boldsymbol{x}}_q) : \mathbf{T}_\ell(\hat{\boldsymbol{x}}_q)\right)$ where the $(\hat{\boldsymbol{x}}_q)$, $1 \le q \le n_Q$ are the so-called Gauss quadrature points and $(w_q) \in \mathbb{R}^{n_Q}$ are their corresponding weights. Similarly, let $\boldsymbol{\sigma}_h \in T_h$ such that $\boldsymbol{\sigma}_h(\hat{\boldsymbol{x}}_q) = \mathbf{0}$ $\forall 1 \le q \le n_Q$; then $\forall \boldsymbol{\tau}_h \in T_h$, $\int_\Omega \boldsymbol{\sigma}_h : \boldsymbol{\tau}_h = 0$. Hence, enforcing

$$(\boldsymbol{\gamma}_h(\hat{\boldsymbol{x}}_q), \boldsymbol{\lambda}_h(\hat{\boldsymbol{x}}_q)) \in \mathscr{DP}(\mu, \mathrm{Bi}, \zeta) \quad \forall 1 \le q \le n_Q$$

implies that any function $\boldsymbol{\tau}_h \in T_h$ interpolating $f_{\mathrm{DS}}(\boldsymbol{\gamma}_h, \boldsymbol{\lambda}_h)$ at the quadrature points $(\hat{\boldsymbol{x}}_q)$ must be zero.

As shown below, defining $(\check{\boldsymbol{x}})_q$, the set of points at which the constraint is enforced, as the set of quadrature points $(\hat{\boldsymbol{x}}_q)$ allows us to retrieve a symmetric Delassus operator.

Recall that for $\boldsymbol{\tau}$ a symmetric tensor field in $T(\Omega)$, $\boldsymbol{\tau}_h \in T_h$ corresponds to its discretized version interpolating the values at the n degrees of freedom $(\boldsymbol{y}_i)$, with $\boldsymbol{\tau}(\boldsymbol{y}_i) = \chi(\underline{\boldsymbol{\tau}}_{[i]})$. Let R be the $\left(\mathfrak{s}_d\, n_Q \times \mathfrak{s}_d\, n\right)$ matrix mapping $\underline{\boldsymbol{\tau}}$ to the interpolated values $\hat{\underline{\boldsymbol{\tau}}}$ at the quadrature points $(\hat{\boldsymbol{x}}_q)$. That is, R is such that $\hat{\underline{\boldsymbol{\tau}}} = R\,\underline{\boldsymbol{\tau}}$, with $\hat{\underline{\boldsymbol{\tau}}} := (\hat{\underline{\boldsymbol{\tau}}}_{[q]})_{1\le q\le n_Q} := \chi^{-1}\left(\boldsymbol{\tau}(\hat{\boldsymbol{x}}_q)\right)$. The matrix R contains $n_Q \times n$ square blocks $\mathfrak{R}_{q,j}$ of size $\mathfrak{s}_d \times \mathfrak{s}_d$ with $\mathfrak{R}_{q,p} = \omega_i^{\boldsymbol{\tau}}(\hat{\boldsymbol{x}}_q)\mathbb{I}_{\mathfrak{s}_d}$ for all $1 \le q \le n_Q$ and $1 \le i \le n$. This translates into the following coefficient-wise expression

$$R_{(q-1)\mathfrak{s}_d+p,(i-1)\mathfrak{s}_d+j} = \omega_i^{\boldsymbol{\tau}}(\hat{\boldsymbol{x}}_q)\delta_{p,j} \quad \text{for} \quad 1 \le p, \ell \le \mathfrak{s}_d. \tag{6.34}$$

Proposition 6.3. *Let $\hat{\underline{\boldsymbol{\lambda}}} := R\,\underline{\boldsymbol{\lambda}}$, and let $R^\dagger$ denote the Moore–Penrose pseudoinverse of R. Then Equations (6.13a — 6.13c) can be discretized as:*
Find $\underline{\boldsymbol{u}} \in \mathbb{R}^{\mathfrak{v}}$, $\hat{\underline{\boldsymbol{\lambda}}}, \hat{\underline{\boldsymbol{\gamma}}} \in \mathbb{R}^{n_Q\,\mathfrak{s}_d}$,

$$\begin{cases} A\underline{\boldsymbol{u}} = B^{\mathsf{T}}R^\dagger\hat{\underline{\boldsymbol{\lambda}}} + \underline{\boldsymbol{l}}^{tot} & \text{(6.35a)} \\ \hat{\underline{\boldsymbol{\gamma}}} = R^{\dagger,\mathsf{T}}B\,\underline{\boldsymbol{u}} + R^{\dagger,\mathsf{T}}\underline{\boldsymbol{k}} & \text{(6.35b)} \\ (\hat{\underline{\boldsymbol{\gamma}}}_{[q]}, \hat{\underline{\boldsymbol{\lambda}}}_{[q]}) \in \mathscr{DP}(\mu, Bi, \zeta) \qquad \forall 1 \le q \le n_Q. & \text{(6.35c)} \end{cases}$$

$\hat{\underline{\boldsymbol{\gamma}}}$ defined as in (6.35b) is such that for $1 \le q \le n_Q$, $\hat{\underline{\boldsymbol{\gamma}}}_{[q]} = 2w_q\boldsymbol{\gamma}_h(\boldsymbol{x}_q)$. The corresponding proof is given in Appendix C.1. This time we have obtained in (6.35) a system which preserves the symmetry of the new Delassus operator $W = R^{\dagger,\mathsf{T}}BA^{-1}B^{\mathsf{T}}R^\dagger$; we will be able to express the solutions to (6.35a–6.35c) as a sequence of convex minimization problems, which are discrete version of (6.18).

Indeed, the convexified version of (6.35a–6.35c), i.e., with $\mu = \zeta$, reduces to

$$A\underline{\boldsymbol{u}} - \underline{\boldsymbol{l}}^{\mathrm{tot}} \in -B^T R^\dagger \mathscr{N}_{\mathscr{K}_{\frac{1}{\mu}}}\left(R^{\dagger,\mathsf{T}}(B\,\underline{\boldsymbol{u}} + \underline{\boldsymbol{k}})\right). \tag{6.36}$$

Under the hypothesis $\exists \underline{\boldsymbol{u}} \in \mathbb{R}^{\mathfrak{v}}, (R^{\dagger,\intercal}(B\,\underline{\boldsymbol{u}} + \underline{\boldsymbol{k}})) \in \operatorname{int} \mathscr{K}_{\frac{1}{\mu}}$, the convexified problem (6.36) reads as the optimality conditions of the minimization problem

$$\min_{\underline{\boldsymbol{u}} \in C} J(\underline{\boldsymbol{u}}), \tag{6.37}$$

$$J(\underline{\boldsymbol{u}}) := \frac{1}{2}\underline{\boldsymbol{u}}^{\intercal} A \underline{\boldsymbol{u}} + \underline{\boldsymbol{u}}^{\intercal} \underline{\boldsymbol{l}} + \mathrm{Bi} \sum_{q=1}^{n_Q} \left\| \hat{\underline{\boldsymbol{\gamma}}}(\underline{\boldsymbol{u}})_{\mathrm{q}\mathbf{T}} \right\|$$

$$\hat{\underline{\boldsymbol{\gamma}}}(\underline{\boldsymbol{u}}) := R^{\dagger,\intercal} B\,\underline{\boldsymbol{u}} + R^{\dagger,\intercal}\,\underline{\boldsymbol{k}}$$

$$C := \left\{ \underline{\boldsymbol{u}} \in \mathbb{R}^{\mathfrak{v}}, \hat{\underline{\boldsymbol{\gamma}}}(\underline{\boldsymbol{u}}) \in \mathscr{K}_{\frac{1}{\mu}} \right\}.$$

Problem (6.37) corresponds to a discretization of the continuous minimization problem (6.18) where the Dirichlet boundary conditions are enforced intrinsically and the strain rate $\mathrm{D}(\mathbf{u})$ is evaluated at the Gauss quadrature points. Indeed,

$$\sum_q \|\hat{\underline{\boldsymbol{\gamma}}}(\underline{\boldsymbol{u}})_{\mathrm{q}\mathbf{T}}\| = \sum_q |2 w_q \boldsymbol{\gamma}_h(\hat{\boldsymbol{x}}_q)_{\mathbf{T}}| = 2 \sum_q w_q |\Pi_{T_h}(\mathrm{D}(\mathbf{u}_h)_{\mathbf{T}})(\hat{\boldsymbol{x}}_q)| \sim 2 \int_\Omega |\mathrm{D}(\mathbf{u}_h)_{\mathbf{T}}|.$$

One remaining difficulty stems from the presence of the matrix $R^{\dagger}$, which in the general case could substantially increase the cost of solving the system. We present in below a few cases for which this difficulty vanishes.

6.3.4 Considerations on $R^{\dagger}$

The first observation is that if the quadrature points $(\hat{\boldsymbol{x}}_q)$ were to coincide with the degrees of freedom $(\boldsymbol{y}_i)$, R would boil down to the identity matrix and the operator $R^{\dagger}$ would not induce any additional cost. This is actually always the case for a piecewise constant ($\mathbb{P}_0$) approximation, for which both the degrees of freedom and the Gauss quadrature points are located at the barycenter of each element.

Trapezoidal quadrature rule For higher-order polynomial basis functions, having the $(\hat{\boldsymbol{x}})_q$ coincide with the $(\boldsymbol{y})_i$ amounts to computing $m(\boldsymbol{\gamma}, \boldsymbol{\tau})$ using a trapezoidal integration rule. Obviously, such an approximation induces a loss of precision — the integral being exact only for functions that are linear between the degrees of freedom. This means that the order of convergence will not increase with that of the basis functions, and using high-order discretization space ($\mathbb{P}_2$ or higher-order polynomials) would be wasteful. However, for piecewise linear ($\mathbb{P}_1$) polynomials, we found this approximation to be acceptable, and used it in practice.

Mixed finite elements In the case of piecewise-polynomial discontinuous basis functions, degrees of freedom are not shared between adjacent elements. When considering such a discretization of the space $T(\Omega)$, the matrix R becomes block-diagonal, and consequently its pseudo-inverse has a similar structure and is easy to compute. The additional cost induced by the presence of the linear operator $R^{\dagger}$ in Problem (6.35) is therefore once again negligible.

Factorization of the B matrix If the discrete velocity space $V_h(\mathbf{0})$ is such that $\mathrm{D}(\mathbf{u}_h)$ is piecewise-polynomial of order less than that of T_h, then the same quadrature rule can be used to compute the bilinear form b. That is, $b(\mathbf{T}_k, \mathbf{V}_j) = \sum_q w_q(\mathbf{T}_k)(\hat{\boldsymbol{x}}_q) : \mathrm{D}(\mathbf{V}_j)(\hat{\boldsymbol{x}}_q)$, and thus $B = R^{\intercal} H$, with H a $n_Q \mathfrak{s}_d \times \mathfrak{v}$ matrix given by

$$H_{(q-1)\mathfrak{s}_d + p, j} := 2 w_q \left[\chi^{-1}\left(\mathrm{D}(\mathbf{V}_j)\right)(\hat{\boldsymbol{x}}_q) \right]_p.$$

Proof.

$$
\begin{aligned}
(R^\intercal H)_{i\,\mathfrak{s}_d+k,j} &= \sum_{q=1}^{n_Q}\sum_{p=1}^{\mathfrak{s}_d} R_{(q-1)\,\mathfrak{s}_d+p,(i-1)\,\mathfrak{s}_d+k} H_{(q-1)\,\mathfrak{s}_d+p,j} \\
&= \sum_{q=1}^{n_Q}\sum_{p=1}^{\mathfrak{s}_d} 2w_q \alpha_i(\hat{\boldsymbol{x}}_q)\delta_p^k \left[\chi^{-1}\left(\mathrm{D}(\mathbf{V}_j)(\hat{\boldsymbol{x}}_q)\right)\right]_p \\
&= \sum_{q=1}^{n_Q} 2w_q \alpha_i(\hat{\boldsymbol{x}}_q)\left\langle \chi(\boldsymbol{e}_k), \mathrm{D}(\mathbf{V}_j)(\hat{\boldsymbol{x}}_q)\right\rangle \\
&= \sum_{q=1}^{n_Q} w_q \mathbf{T}_{(i-1)\,\mathfrak{s}_d+k}(\hat{\boldsymbol{x}}_q) : \mathrm{D}(\mathbf{V}_j)(\hat{\boldsymbol{x}}_q) = b(\mathbf{T}_{(i-1)\,\mathfrak{s}_d+k}, \mathbf{V}_j) = B_{(i-1)\,\mathfrak{s}_d+k,j}.
\end{aligned}
$$

□

In this case, a discrete version of (6.13a–6.13c) can be written simply as:
Find $\underline{\boldsymbol{u}} \in \mathbb{R}^{\mathfrak{v}}$, $\underline{\hat{\boldsymbol{\lambda}}}, \underline{\hat{\boldsymbol{\gamma}}} \in \mathbb{R}^{n_Q\,\mathfrak{s}_d}$,

$$
\begin{cases}
A\underline{\boldsymbol{u}} = H^\intercal \underline{\hat{\boldsymbol{\lambda}}} + \underline{\boldsymbol{l}}^{\mathrm{tot}} \\
\underline{\hat{\boldsymbol{\gamma}}} = H\,\underline{\boldsymbol{u}} + \underline{\hat{\boldsymbol{k}}} \\
(\underline{\hat{\boldsymbol{\gamma}}}_{[q]}, \underline{\hat{\boldsymbol{\lambda}}}_{[q]}) \in \mathscr{D}\mathscr{P}(\mu, \mathrm{Bi}, \zeta) \quad \forall 1 \le q \le n_Q,
\end{cases}
$$

where

$$
\underline{\hat{\boldsymbol{k}}}_{q\,\mathfrak{s}_d+p} := 2w_q \left[\chi^{-1}\left(\mathrm{D}(\mathbf{u}^*_{D,h})(\hat{\boldsymbol{x}}_q)\right)\right]_p,
$$

and the pseudo-inverse needs not be computed. However, note that unlike the previous methods which can be used in conjunction with black-box finite-element libraries, this approach requires being able to modify the assembly of matrice corresponding to the "b" bilinear form.

6.3.5 Final discrete system

For brevity of notation and since there are no more ambiguities, from now on we shall drop the decorations of the variables, i.e., we shall consider Problem (6.35) written as:
Find $\boldsymbol{u} \in \mathbb{R}^{\mathfrak{v}}$, $\boldsymbol{\lambda}, \boldsymbol{\gamma} \in \mathbb{R}^{n_Q\,\mathfrak{s}_d}$,

$$
\begin{cases}
A\boldsymbol{u} = B^\intercal R^\dagger \boldsymbol{\lambda} + \boldsymbol{l} \\
\boldsymbol{\gamma} = R^{\dagger,\intercal} B\,\boldsymbol{u} + R^{\dagger,\intercal}\boldsymbol{k} \\
(\boldsymbol{\gamma}_i, \boldsymbol{\lambda}_i) \in \mathscr{D}\mathscr{P}(\mu_i, \mathrm{Bi}_i, \zeta_i) \quad \forall 1 \le i \le n_Q.
\end{cases}
\tag{6.38}
$$

Under this form, the similarity of (6.38) with the DCFP (2.17) is clear. There remains only two differences:

- The constraint is more general; we may have to adapt our DCFP solvers to handle the dilatancy ζ and a truncated SOC when $\mathrm{Bi} > 0$;
- In contrast with DEM methods where the global stiffness matrix was block-diagonal, and similarly to the case of cloth dynamics, here A is sparse but with a dense inverse.

Discussion on the discretization strategy At no point have we attempted to study the rate of convergence of our finite-element discretization; in particular, we did not specify any *inf-sup*-like condition. It is probable that one should be required; consider homogeneous Dirichlet boundary conditions, then our unilateral incompressibility constraint becomes equivalent to full incompressibility. To mitigate this consideration, in our Results section we present convergence

tests on simple Bingham Poiseuille flows for which we have an analytical solution, and observe convergence of the velocity solution in L_2- and H^1-norms.

The discrete setting will also allow us to formulate stronger existence properties than for the continuous problem, using the criterion from of Cadoux (2009).

6.4 Solving the discrete problem

In this section, we are concerned with the numerical resolution of System (6.38).

6.4.1 Discrete Cadoux fixed-point algorithm

We have already hinted in the previous section that the solutions to System (6.38) can be characterized as the fixed points of a sequence of minimization problems structurally similar to Problem (6.37); we demonstrate this assertion below.

Indeed, let S denote the subspace of $R^{n_Q\mathfrak{s}_d}$ with vanishing tangential components, $S = \{\boldsymbol{s} \in \mathbb{R}^{n_Q\mathfrak{s}_d},\ \boldsymbol{s}_{\mathrm{iT}} = \boldsymbol{0}\ \forall 1 \leq i \leq n_Q\}$, and S_+ the positive subset of S, $S_+ = \{\boldsymbol{s} \in S, \boldsymbol{s}_{\mathrm{iN}} \geq 0\ \forall 1 \leq i \leq n_Q\}$. For any $\boldsymbol{s} \in S$, consider the minimization problem (6.39),

$$
\begin{aligned}
q(\boldsymbol{s}) &:= \min_{\boldsymbol{u}\in C(\boldsymbol{s})} J(\boldsymbol{u}), \qquad (6.39)\\
J(\boldsymbol{u}) &:= \frac{1}{2}\boldsymbol{u}^{\intercal}A\boldsymbol{u} + \boldsymbol{u}^{\intercal}\boldsymbol{l} + \mathrm{Bi}\sum_{i=1}^{n_Q}\|\boldsymbol{\gamma}(\boldsymbol{u})_{\mathrm{iT}}\|\\
\boldsymbol{\gamma}(\boldsymbol{u}) &:= R^{\dagger,\intercal}B\,\boldsymbol{u} + R^{\dagger,\intercal}\boldsymbol{k}\\
C(\boldsymbol{s}) &:= \left\{\boldsymbol{u} \in \mathbb{R}^{\mathfrak{v}}, \boldsymbol{\gamma}(\boldsymbol{u}) + \boldsymbol{s} \in \mathscr{K}_{\frac{1}{\mu}}\right\}.
\end{aligned}
$$

Proceeding as in the discrete DEM (Section 2.3.2) or continuous FEM (Section 6.2.3) cases, we can show that any $\boldsymbol{u} \in \mathbb{R}^{\mathfrak{v}}$ satisfying (6.40),

$$
A\bar{\boldsymbol{u}} - \boldsymbol{l} \in -B^T R^{\dagger}\left[\mathscr{N}_{\mathscr{K}_{\frac{1}{\mu}}}(\boldsymbol{\gamma}(\boldsymbol{u}) + \boldsymbol{s}) + \mathrm{Bi}\partial\|\cdot_{\mathrm{T}}\|(\boldsymbol{\gamma}(\boldsymbol{u}))\right] \tag{6.40}
$$

is a solution of (6.39). This sufficient condition is also necessary when $\mathscr{H}(\underline{\boldsymbol{s}})$ holds,

$$
\exists \boldsymbol{u} \in \mathbb{R}^{\mathfrak{v}}, \boldsymbol{\gamma}(\boldsymbol{u}) + \boldsymbol{s} \in \operatorname{int}\mathscr{K}_{\frac{1}{\mu}}. \tag{$\mathscr{H}(\underline{\boldsymbol{s}})$}
$$

Moreover, Equation (6.40) can be equivalently rewritten as

$$
\begin{cases}
A\boldsymbol{u} = B^{\intercal}R^{\dagger}\,\boldsymbol{\lambda} + \boldsymbol{l}\\
\boldsymbol{\gamma} = R^{\dagger,\intercal}B\,\boldsymbol{u} + R^{\dagger,\intercal}\boldsymbol{k}\\
\boldsymbol{\lambda}_i \in -\mathscr{N}_{\mathscr{K}_{\frac{1}{\mu_i}}}(\boldsymbol{\gamma}_i + \boldsymbol{s}_i) - \mathrm{Bi}\partial\|\cdot_{\mathrm{T}}\|(\boldsymbol{\gamma}_i) \quad \forall 1 \leq i \leq n_Q.
\end{cases} \tag{6.41}
$$

Using once again Equation (1.25) to rewrite the last line of (6.41),

$$
\begin{aligned}
\boldsymbol{\lambda}_i \in -\mathscr{N}_{\mathscr{K}_{\frac{1}{\mu_i}}}(\boldsymbol{\gamma}_i + \boldsymbol{s}_i) - \mathrm{Bi}\partial\|\cdot_{\mathrm{T}}\|(\boldsymbol{\gamma}_i) &\iff \boldsymbol{\lambda}_i \in -\partial\left(\mathscr{I}_{-\mathscr{T}_{\mu_i,\mathrm{Bi}}}{}^{\star}\right)(\boldsymbol{\gamma}_i + \boldsymbol{s}_i)\\
&\iff \boldsymbol{\gamma}_i + \boldsymbol{s}_i \in \mathscr{N}_{-\mathscr{T}_{\mu_i,\mathrm{Bi}}}(-\boldsymbol{\lambda}),
\end{aligned}
$$

and recalling the definition (1.19) of the Drucker–Prager flow rule, one can check that Equation (6.40) coincide with System (6.38) when $\boldsymbol{s}_{\mathrm{iN}} = (\mu - \zeta)\|\boldsymbol{\gamma}(\boldsymbol{u})_{\mathrm{iT}}\|$ for any $1 \leq i \leq n_Q$. In other words, we can once again solve our discrete flow problem (6.38) using a fixed point algorithm.

Property 6.3 (Cadoux fixed-point algorithm)**.** *Let $u : S_+ \to \mathbb{R}^{\mathfrak{v}}$ be a function associating to each $\boldsymbol{s} \in S_+$ a solution to the minimization problem (6.39), i.e., $u(\boldsymbol{s}) \in C(\boldsymbol{s})$ and $J(u(\boldsymbol{s})) = q(\boldsymbol{s})$. Let $s : \mathbb{R}^{\mathfrak{v}} \to S_+$ such that $\forall \boldsymbol{u} \in \mathbb{R}^{\mathfrak{v}}$, $\forall 1 \leq i \leq n_Q$, $s(\boldsymbol{u})_{iN} = (\mu - \zeta)\|\boldsymbol{\gamma}(\boldsymbol{u})_{iT}\|$.*

The Cadoux fixed-point algorithm*, induced by $\boldsymbol{s}^{k+1} := s \circ u(\boldsymbol{s}^k)$ and an initial guess $\boldsymbol{s}^0 \in S_+$, will be well-defined as long as $C(\mathbf{0}) \neq \mathbf{0}$ and $Re < +\infty$.*

Moreover, let $\bar{\boldsymbol{s}}$ be a fixed-point of $s \circ u$ satisfying $\mathscr{H}(\bar{\boldsymbol{s}})$; then the minimization problem $q(\bar{\boldsymbol{s}})$ defined in (6.39) admits a unique solution, and its optimality conditions (6.41) realize a solution ($\bar{\boldsymbol{u}}$, $\bar{\boldsymbol{\lambda}}$, $\bar{\boldsymbol{\gamma}}$) of the discrete flow problem (6.38).

Indeed, if $C(\mathbf{0})$ is non-empty, then all the minimization problems in the sequence generated by the fixed-point algorithm are feasible. If $\mathrm{Re} < +\infty$, the objective function is strictly convex and coercive, hence each intermediate problem admits a unique solution, and the function u is well-defined.

The following existence property can also be directly adapted from (Cadoux 2009, Theorem 3.19).

Property 6.4 (Cadoux existence criterion)**.** *If $C(\mathbf{0})$ is non-empty, then the Cadoux fixed point algorithm defined in Property 6.3 admits a fixed-point $\bar{\boldsymbol{s}}$.*

Indeed, while Cadoux (2009) did not treat the case $\mathrm{Bi} > 0$, as the strict convexity, coercivity and continuity of J are preserved, his analysis holds nevertheless without modifications.

Homogeneous Dirichlet boundary conditions Let us emphasize once again that this existence criterion is only sufficient, and not necessary. In the case $\mu = 0$, the strong hypothesis $\mathscr{H}(\mathbf{0})$ amounts to the existence of a velocity field with strictly positive divergence everywhere – this requires outwards Dirichlet boundary conditions, or a Neumann boundary. The weaker hypothesis in Property 6.4 only requires that there exists a velocity field with nowhere strictly negative divergence. Homogeneous Dirichlet boundary conditions only satisfy the latter, weaker criterion $C(\mathbf{0}) \neq \emptyset$. However we can always exhibit a trivial solution in $\boldsymbol{u}, \boldsymbol{\lambda}$; for $\mu = 0$, it corresponds to the incompressible Stokes solution, and for $\mu > 0$, to the null velocity solution, $\boldsymbol{u} = 0$.

Proof. Let us use our original notations, with the underline to differentiate the concatenated coefficient vectors from the discretized fields. By construction, we have $\underline{\boldsymbol{k}} = 0$. Moreover, the homogeneous Dirichlet condition imposes $\int_\Omega \nabla \cdot \mathbf{u}_h = \int_{\partial\Omega} \mathbf{u}_{D,h} \cdot \boldsymbol{n}_\Omega = 0$. As the rheology requires $\nabla \cdot \mathbf{u}_h \geq 0$, velocity solutions must have null divergence, which means $(R^{\dagger,\intercal} B \underline{\boldsymbol{u}})_{\mathrm{N}} = \mathbf{0}$.

In the case $\mu = 0$, $\bar{\boldsymbol{s}} := \mathbf{0}$ is a fixed-point of $s \circ u$, and the primal feasibility condition $\underline{\boldsymbol{u}} \in C(\mathbf{0})$ boils down to $(R^{\dagger,\intercal} B \underline{\boldsymbol{u}})_{\mathrm{N}} = \mathbf{0}$. The optimality condition reads

$$-\nabla J \in \mathscr{N}_{C(\mathbf{0})} \underline{\boldsymbol{u}} = (\mathrm{Ker}(R^{\dagger,\intercal} B)_{\mathrm{N}})^\perp = \mathrm{Im}(R^{\dagger,\intercal} B)_{\mathrm{N}}^{\intercal},$$

so there exists $\underline{\boldsymbol{\lambda}} \in \mathbb{R}^{n_Q \mathfrak{s}_d}$ with $\nabla J = B^T R^\dagger \underline{\boldsymbol{\lambda}}$ and $\underline{\boldsymbol{\lambda}}_{\mathbf{T}} = 0$.

In the case $\mu > 0$, $\underline{\boldsymbol{u}} \in C(\mathbf{0})$ boils down to $R^{\dagger,\intercal} B \underline{\boldsymbol{u}} = \mathbf{0}$. This means that the solution $u(\mathbf{0})$ of the primal problem at $\boldsymbol{s} = \mathbf{0}$ satisfies $\|(B^\intercal R^\dagger \underline{\boldsymbol{u}})_{\mathbf{T}}\| = 0$, and therefore $\bar{\boldsymbol{s}} := \mathbf{0}$ is a fixed point of $s \circ u$. The optimality condition yields $\exists \underline{\boldsymbol{\lambda}} \in \mathbb{R}^{n_Q \mathfrak{s}_d}$, $\nabla J = B^T R^\dagger \underline{\boldsymbol{\lambda}}$.

In both cases, we define $\boldsymbol{\lambda}_h^* := \boldsymbol{\lambda}_h + \Delta_p \boldsymbol{\iota}_d$, with Δ_p constant over all discretization points. By construction, $\underline{\boldsymbol{\lambda}} - \underline{\boldsymbol{\lambda}}^* \in \mathrm{Ker}\, B^T R^\dagger$. We can choose Δ_p such that $\underline{\boldsymbol{\lambda}}^* \in \mathscr{K}_\mu$; for $\mu = 0$, it suffices to take $\Delta_p := -\min_i \underline{\boldsymbol{\lambda}}_{\mathrm{iN}}$, and for $\mu > 0$, $\Delta_p := -\min_i(\underline{\boldsymbol{\lambda}}_{\mathrm{iN}} - \frac{1}{\mu}\|\underline{\boldsymbol{\lambda}}_{\mathrm{iT}}\|)$. We have then $A\underline{\boldsymbol{u}} - \underline{\boldsymbol{l}} = \nabla J = B^T R^\dagger \underline{\boldsymbol{\lambda}}^*$, and it can be easily verified that in both cases $\underline{\boldsymbol{\gamma}}^\intercal \underline{\boldsymbol{\lambda}}^* = 0$. □

6.4.2 Dual problem

If the matrix A is invertible (which, once again, is achieved as soon as $\mathrm{Re} < +\infty$) then the velocity variable $\boldsymbol{u}$ can be eliminated in System (6.41), so that the latter system can be written equivalently as

$$\begin{cases} \boldsymbol{\gamma} = W\boldsymbol{\lambda} + \boldsymbol{b} \\ \boldsymbol{\gamma}_i + \boldsymbol{s}_i \in -\mathcal{N}_{\mathscr{T}_{\mu_i,\mathrm{Bi}}}(\boldsymbol{\lambda}_i) \quad \forall 1 \le i \le n_Q, \end{cases} \tag{6.42}$$

where $W := R^{\dagger,\intercal} B A^{-1} B^\intercal R^\dagger$ and $\boldsymbol{b} := R^{\dagger,\intercal}\left(\boldsymbol{k} + BA^{-1}\boldsymbol{l}\right)$. As W is symmetric positive semi-definite, we recognize (6.42) as the optimality conditions of the minimization problem (6.43),

$$\min_{\boldsymbol{\lambda}\in\mathscr{T}_{\mu,\mathrm{Bi}}} \frac{1}{2}\boldsymbol{\lambda}^\intercal W\boldsymbol{\lambda} + (\boldsymbol{b}+\boldsymbol{s})^\intercal\boldsymbol{\lambda}. \tag{6.43}$$

Problem (6.43) is the dual (in the sense of Fenchel) of the primal minimization problem (6.39). The Cadoux fixed-point algorithm defined in Property 6.3 can thus be iterated by solving the intermediate problems under either primal or dual form; the next section is dedicated to the numerical resolution of those minimization problems.

6.4.3 Solving the minimization problems

Problems (6.39) and (6.43) are very close to SOCQP (and actually are SOCQP when $\mathrm{Bi} = 0$); for the primal, the difference is just a modification of the objective function (which is no longer quadratic), and for the dual, only the constraint is modified by replacing the SOC with a truncated SOC. It is therefore likely that the algorithms presented in Chapter 3 for solving SOCQP can be easily adapted to our flow problems. However, remember that the inverse of our matrix A is dense, therefore we will not be able to use the Gauss–Seidel solver. Moreover, as we target relatively large meshes, we will not consider performing directly root-finding methods on complementarity functions. This leaves us with two strategies, interior-points (Section 3.2) and proximal algorithms (Section 3.3).

Interior points When $\mathrm{Bi} = 0$ the SOCP formulations of the primal and dual problems given in Section 3.2 can directly be used. Moreover, when $\mathrm{Bi} > 0$, our modified minimization problems (6.39) and (6.43) can still be put under SOCP form.

Indeed, the "Bi" part of our primal objective function can again be transformed into a linear contribution with SOC constraints using a trick similar to that of the quadratic part. Indeed, $\min_{\boldsymbol{x}\in\mathbb{R}^{nd}} \sum_i \|\boldsymbol{x}_i\|$ can be rewritten as $\min_{\boldsymbol{r}\in\mathbb{R}^n \boldsymbol{r}_i}$ such that $\boldsymbol{r}_i \ge \|\boldsymbol{x}_i\|\ \forall i$. Let L be a square-root of A, i.e., $A = LL^\intercal$, then a solution of the primal minimization problem (6.39) can be found by solving the SOCP

$$\begin{cases} \min\limits_{\boldsymbol{u}\in\mathbb{R}^v, t\in\mathbb{R}, \boldsymbol{r}\in R^{n_Q}} t + \mathrm{Bi}\sum\limits_{i=1}^{n_Q} \boldsymbol{r}_i - \boldsymbol{u}^\intercal\boldsymbol{l} \\ \boldsymbol{z} = L^\intercal\boldsymbol{u} \\ R^\intercal\boldsymbol{\gamma} = B\boldsymbol{u} + \boldsymbol{k} + R^\intercal\boldsymbol{s} \\ \boldsymbol{\gamma}_i \in \mathscr{K}_{\frac{1}{\mu_i}} \qquad \forall i = 1\dots n_Q \\ (\boldsymbol{r}_i, \boldsymbol{\gamma}_{i\mathrm{T}}) \in \mathscr{K}_1 \qquad \forall i = 1\dots n_Q \\ (1, t, \boldsymbol{z}) \in \mathscr{R}\mathscr{K}. \end{cases} \tag{6.44}$$

The objective function of the dual minimization problem has not changed, however the constraint on $\boldsymbol{\lambda}$ is no longer conical. Yet this constraint can still be expressed using two SOC constraints:

$$\boldsymbol{\lambda} \in \mathscr{T}_{\mu,\mathrm{Bi}} \iff \begin{cases} \boldsymbol{\lambda} = \boldsymbol{\lambda}_K + \boldsymbol{\lambda}_B \\ \boldsymbol{\lambda}_{Ki} \in \mathscr{K}_{\mu_i} \qquad \forall 1 \le i \le n_Q \\ \boldsymbol{\lambda}_{B\mathrm{iT}} \le \mathrm{Bi} \text{ and } \boldsymbol{\lambda}_{B\mathrm{iN}} = 0 \quad \forall 1 \le i \le n_Q \end{cases}$$

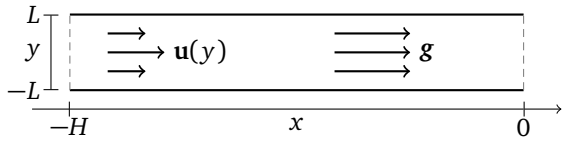


Figure 6.2: *2D channel for the Bingham Poiseuille flow*

Using this insight, we can propose another SOCP for the dual minimization problem (6.43),

$$\begin{cases} \min\limits_{\boldsymbol{\lambda}_K,\boldsymbol{\lambda}_B\in\mathbb{R}^{n_Q\, s_d},t\in\mathbb{R}} t+(\boldsymbol{\lambda}_B+\boldsymbol{\lambda}_K)^\intercal(\boldsymbol{b}+\boldsymbol{s}) & \\ R\boldsymbol{y}=\boldsymbol{\lambda}_K+\boldsymbol{\lambda}_B & \\ L^\intercal\boldsymbol{z}=H^\intercal\boldsymbol{y} & \\ \boldsymbol{\lambda}_{Ki}\in\mathscr{K}_{\mu_i} & \forall i=1\dots n_Q \\ (\mathrm{Bi},\boldsymbol{\lambda}_{B\mathrm{iT}})\in\mathscr{K}_1 & \forall i=1\dots n_Q \\ \boldsymbol{\lambda}_{B\mathrm{iN}}=0 & \forall i=1\dots n_Q \\ (1,t,\boldsymbol{z})\in\mathscr{R}\mathscr{K}. & \end{cases} \tag{6.45}$$

Proximal algorithms As computing an orthogonal projection on the truncated cone $\mathscr{T}_{\mu,\mathrm{Bi}}$ is not more complex than for $\mathscr{K}_\mu$, the projected gradient descent, projected-gradient algorithms and their variants (Section 3.3.2) for solving the dual minimization problem (2.21) can be trivially adapted to our modified dual (6.43). Note that as we do not want to explicitly assemble the Delassus operator W, each step of these algorithms will involve solving a linear system with matrix A. This can be done efficiently by precomputing a Cholesky factorization of A, or by using a truncated conjugate-gradient algorithm.

6.5 Results

All the finite element simulations presented in this section were performed using the open-source library Rheolef (Saramito 2015), and were run on a quad-core Intel(R) Xeon W3520 machine with 8GB memory.

6.5.1 Model problems

In this section, we validate our method on simple problems for which we can derive analytic solutions.

Bingham Poiseuille flow

We consider a Bingham Poiseuille flow as illustrated in Figure 6.2, with the following boundary conditions,

$$\begin{aligned} \mathbf{u}(x,-L)=\mathbf{u}(x,L)=0\ \forall x\in[-H,0] \\ \mathbf{u}(-H,y)=u_x(-H,y)\boldsymbol{e}_x\ \forall y\in[-L,L] \\ \mathbf{u}(0,y)=u_x(0,y)\boldsymbol{e}_x\ \forall y\in[-L,L]. \end{aligned}$$

Figure 6.3 shows velocity profiles for different values of Bi, for both the analytic solution and our model using a $\mathbb{P}_4-\mathbb{P}_{1d}$ approximation. Using a nonsmooth solver allows us to recover the correct profile even for high values of Bi without any parameter tuning.

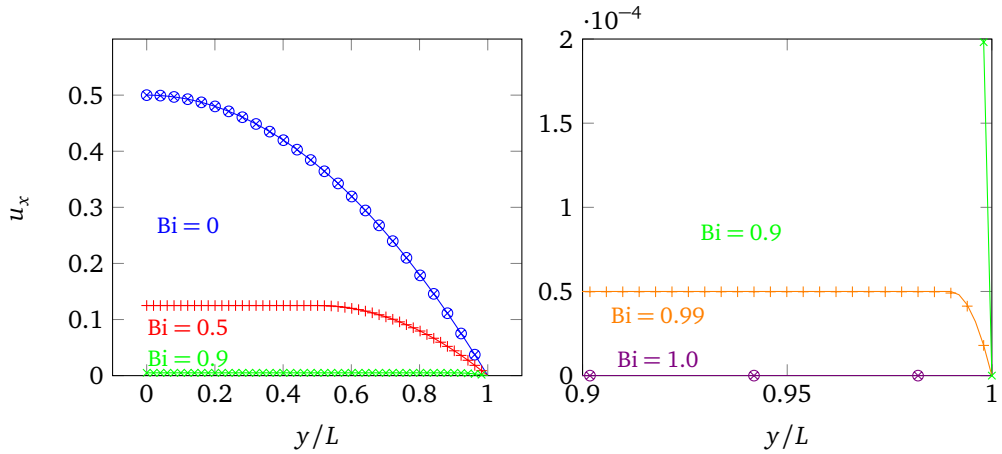


Figure 6.3: *Comparison of the steady velocity profiles $u_x(-H/2, y)$ between our numerical model using a $\mathbb{P}_2-\mathbb{P}_{1d}$ approximation (marks), and the analytic Bingham Poiseuille flow (lines). Plots for high values of Bi (right) are zoomed in compared to plots for lower Bi values (left).*

Convergence of spatial discretization We study how the error between our method and the analytic solutions for Bi $= 0.5$ and Bi $= 0.9$ decreases as we uniformly refine a mesh with initial characteristic edge length h_0. Results for various FEM approximation orders are shown in Figure 6.4. Convergence was observed for all approximation orders, and we found that $\mathbb{P}_2-\mathbb{P}_1$ and $\mathbb{P}_2-\mathbb{P}_{1d}$ achieved a good ratio of convergence speed versus computational cost. Conversely, the higher-order approximation $\mathbb{P}_3-\mathbb{P}_{2d}$ performed relatively poorly on the finer meshes, which we interpret as being the result of numerically more complex quadrature rules. $\mathbb{P}_4-\mathbb{P}_{1d}$, with high-order velocities but low-order stresses, yielded consistently the best results, at the cost of a very large A matrix.

Bagnold profiles

We consider the flow of a granular layer of height $H = 1$ on a rough infinite inclined plane with angle α, as described in (Lagrée et al. 2011) and illustrated in Figure 6.5. We assume the flow to be slow enough to neglect inertial terms.

The conservation of momentum on the longitudinal x-axis and perpendicular y-axis reads

$$\frac{\partial}{\partial y}\left(\eta\frac{\partial \mathbf{u}_x}{\partial y} + \boldsymbol{\tau}_{xy}\right) = -\sin\alpha$$
$$\frac{\partial \mathrm{p}}{\partial y} = -\cos\alpha.$$

Integrating the second equation with the condition that the pressure should be zero at the top of the granular layer, i.e., $\mathrm{p}(H) = 0$, gives $\mathrm{p} = (1-y)\cos\alpha$.

We consider the case of an avalanching flow with $\frac{\partial \mathbf{u}_x}{\partial y} > 0$ for $y < 1$. This means that the friction is saturated, therefore $\tau_{xy} = \mu(1-y)\cos(\alpha)$. We get

$$\frac{\partial}{\partial y}\eta\frac{\partial \mathbf{u}_x}{\partial y} = -\sin\alpha + \mu\cos\alpha.$$

A Neumann boundary condition at the interface $\frac{\partial \mathbf{u}_x}{\partial y}(1) = 0$ imposes $\frac{\partial}{\partial y}\eta\frac{\partial \mathbf{u}_x}{\partial y} < 0$, which means $\mu < \tan\alpha$.

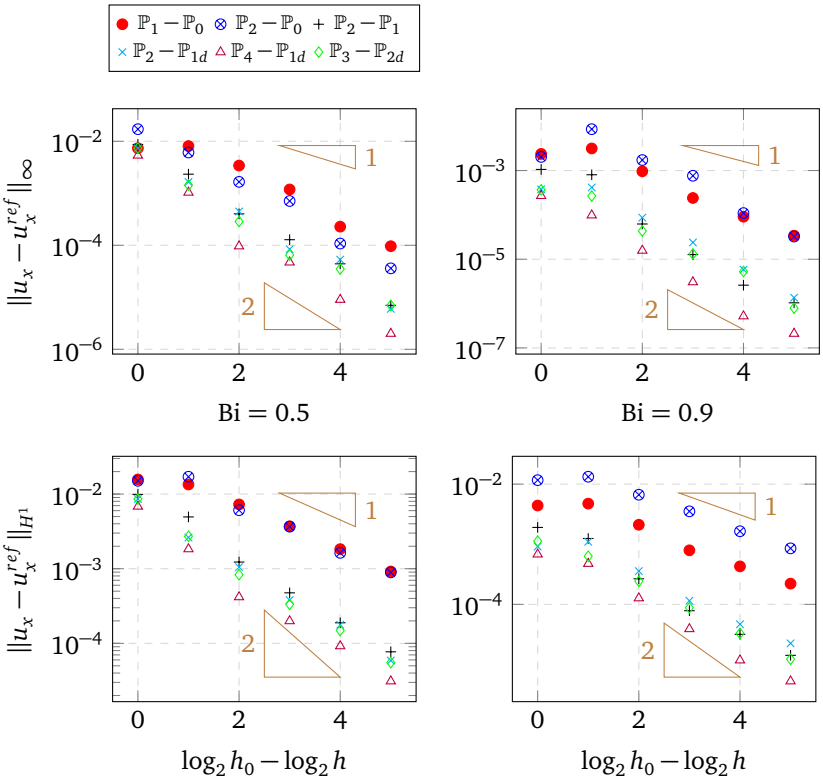


Figure 6.4: *L_∞-norm (top) and H^1-norm (bottom) convergence plots for various FEM approximation orders, for $Bi = 0.5$ (left) and $Bi = 0.9$ (right). u_x^{ref} denotes the analytic solution.*

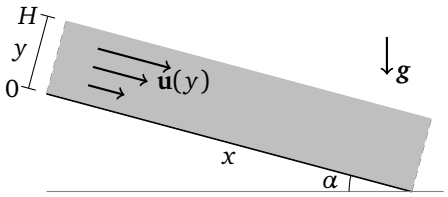


Figure 6.5: *Flow on an infinite inclined plane*

For spatially constant (Newtonian) η and μ, we get $\frac{\partial \mathbf{u}_x}{\partial y} = \frac{1-y}{\eta}\sin\alpha - (\mu\cos\alpha)$, and $\mathbf{u}_x(y)$ is quadratic. In order to retrieve the typical $\frac{3}{2}$ power of the Bagnold profile, we can instead choose $\eta(y) := |\mathrm{D}(\mathbf{u})| = \frac{1}{2}\frac{\partial \mathbf{u}_x}{\partial y}$, which gives the analytic expressions

$$\frac{\partial \mathbf{u}_x}{\partial y} = \sqrt{2\sin\alpha - (\mu\cos\alpha)}\,(1-y)^{\frac{1}{2}}$$
$$\mathbf{u}_x(y) = \frac{2}{3}\sqrt{2\sin\alpha - (\mu\cos\alpha)}\left(1-(1-y)^{\frac{3}{2}}\right).$$

We simulated this model on a square patch using the algorithm presented in Section 6.2. The value of $\boldsymbol{\sigma}^{\text{ext}}$ on the upstream and downstream boundaries was computed using the analytic solution. In order to handle the non-constant η, we used a fixed-point algorithm which happened to converge very fast in practice — a dozen or so of iterations.

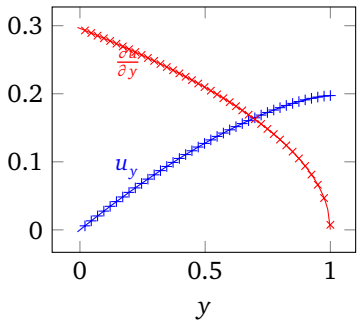


Figure 6.6: *Comparison between our numerical solution (marks) and the analytic one (lines), for the velocity and shear-rate profiles of a Bagnold avalanche flow with* $\mu = 0.5$ *and* $\alpha = \tan^{-1}(1.1\mu)$.

Numerical and analytic profiles are compared in Figure 6.6. Once again, we observe a very good agreement between our simulations and the analytic solution.

6.5.2 Flow around a cylinder

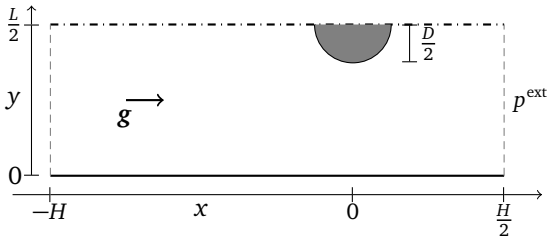


Figure 6.7: *Flow around a cylinder of diameter D in 2D channel of width L. The speed of the flow can be adjusted by varying the external pressure at the downstream boundary* p^{ext}.

In this section, we study a gravity induced flow in a 2D channel of width L around a cylinder of diameter D, as shown in Figure 6.7. We use no-slip boundary conditions for the velocity on the sides of the channel and on the cylinder, and homogeneous Neumann conditions at the upstream and downstream boundaries.

Equivalent drag We attempt to reproduce the experimental setting presented in (Chehata et al. 2003). Using a regularization of the $\mu(I)$ rheology, Chauchat and Médale (2014) performed simulations that showed a good qualitative agreement with the experimental results. However, their approach suffered from two drawbacks:

- The regularization induces a creeping flow even when none should occur;
- The $\nabla \cdot \mathbf{u} = 0$ condition leads to negative pressures behind the obstacle. This is outside the domain of validity of the $\mu(I)$ rheology.

Our method also suffer from approximations; we assume a constant volume fraction of grains, even if this should clearly not be the case behind the obstacle. As the flow is slow, we also neglect its inertia. We use a constant friction coefficient, $\mu = 0.5$, and set $\text{Re} = 100$ and $\text{Bi} = 0$. We vary the average upstream velocity U_∞ by setting different values for the external pressure p^{ext} at the downstream interface; the lower this pressure, the faster the grains will flow. This emulates the varying outlet size in (Chehata et al. 2003). We set no external pressure at the upstream boundary.

The grains exercise a longitudinal drag force on the cylinder O that can be computed as $F_D := \int_{\partial O} \boldsymbol{\lambda n}_O . \boldsymbol{e}_x$. The Froude number is defined from the dimensionless average upstream velocity U_∞ and the length ratio $\tilde{D} := \frac{D}{L}$ as $\mathrm{Fr} := \frac{U_\infty}{\sqrt{\tilde{D}}}$.

The dimensionless equivalent drag coefficient C_D is then deduced as

$$C_D := \frac{F_D}{\frac{1}{2} U_\infty^2 \tilde{D}}.$$

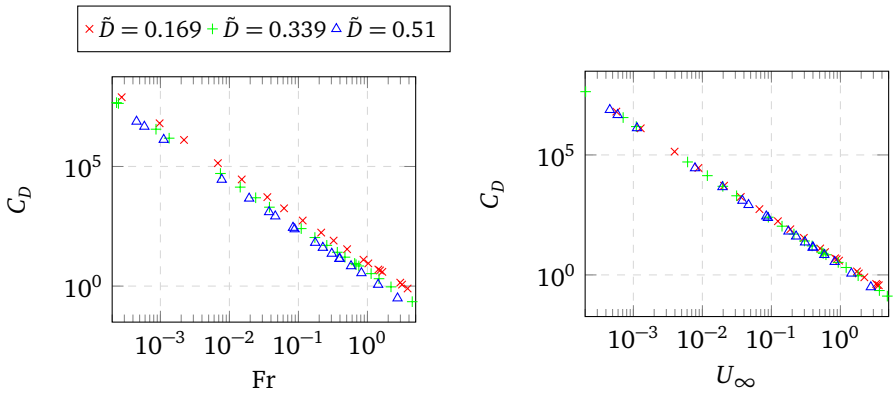


Figure 6.8: *Drag coefficient C_D versus Froude number Fr (left) and upstream velocity U_∞ (right) for the granular flow around a cylinder*

Figure 6.8 shows on a logarithmic scale the evolution of the drag coefficient C_D with the Froude number (left) or average velocity (right), for different cylinder diameters. We retrieve the linear profiles observed in (Chauchat and Médale 2014; Chehata et al. 2003). The fact that the data points for the different diameters become aligned on the right plot means that the slope of $\frac{F_D}{\frac{1}{2}\tilde{D}}$ is independent of $\tilde{D}$; the drag force depends linearly on the diameter of the obstacle.

Visualization of velocity and stress fields We now consider a narrow channel ($L = 4D$) with a free-slip boundary condition on the side walls — results for no-slip walls are depicted in (Daviet and Bertails-Descoubes 2016b, Figure 14). Figure 6.9 collects plots of the velocity and stress fields across the domain. The pressure field is of special interest as it possesses two notable features:

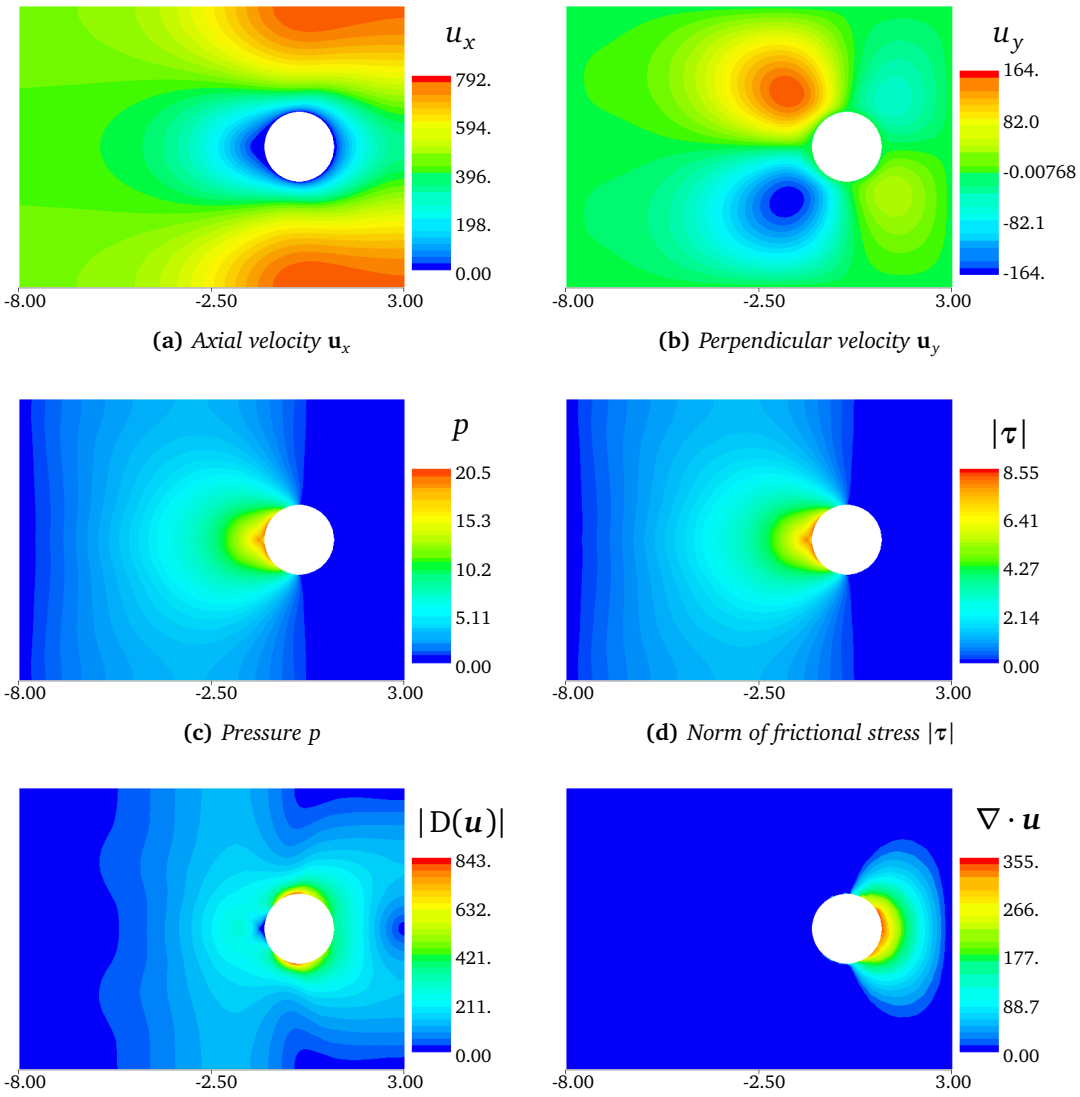
- First, it validates the benefit of allowing the dilation of the flow: the pressure does vanish in the wake of the obstacle, and the pressure intensity corresponds qualitatively with the experimental results of Seguin et al. (2016) using photoelastic grains;
- The zone of highest pressure is not located at the very front of the obstacle. Instead, we observe the formation of high-pressure arch above this point;

The shear rate plot in Figure 6.9(e) highlights the existence of a triangular rigid zone in front of the obstacle.

6.5.3 Extension to inertial flows: discharge of a silo

Beverloo scaling One of the most widely accepted macroscopic feature of the granular flow in a silo is the so-called Berveloo scaling (Beverloo et al. 1961), stating that the discharge rate Q depends on the diameter of the outlet to the power $d - \frac{1}{2}$,

$$Q = C\sqrt{g}\left(L - kD_g\right)^{d-\frac{1}{2}},$$



(a) *Axial velocity* $\mathbf{u}_x$ **(b)** *Perpendicular velocity* $\mathbf{u}_y$

(c) *Pressure* p **(d)** *Norm of frictional stress* $|\tau|$

(e) *Strain rate* $|\mathrm{D}(\mathbf{u})|$ **(f)** *Divergence of velocity* $\nabla \cdot \mathbf{u}$

Figure 6.9: *Velocity and stress fields for the flow around a cylinder in a narrow channel without wall friction, with* $\mu = 0.3$*,* $Re = 100$*,* $Bi = 0$ *and* $\tilde{D} = \frac{1}{4}$

where D_g is the diameter of a grain, and C and k are dimensionless constants depending on the silo geometry and granular properties. The number k typically lies within the range $1 < k < 3$.

The Beverloo phenomenon is particularly relevant for us as it has been shown that such a scaling cannot be recovered for Newtonian flows (Staron et al. 2012), nor for flows with a yield stress that does not depend on the pressure. The physical justification of the scaling involves inertia (Mankoc et al. 2007), so it is hopeless to attempt to retrieve it solely with the formulation of Section 6.2. For this problem, we therefore need to modify our equations to account for the grains inertia.

Modification of the variational formulation We now add an inertial term to our momentum conservation equation (6.10),

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u}\right) - \nabla\cdot\left[2\eta E^b\dot{\boldsymbol{\varepsilon}} - \boldsymbol{\lambda}\right] = \rho g \boldsymbol{e}_g,$$

which can be made dimensionless by defining $\tilde{t}$ such that $t = \frac{L}{U}\tilde{t} = \sqrt{\frac{L}{g}}\tilde{t}$,

$$\left(\frac{\partial \tilde{\mathbf{u}}}{\partial \tilde{t}} + (\tilde{\mathbf{u}}\cdot\tilde{\nabla})\tilde{\mathbf{u}}\right) - \tilde{\nabla}\cdot\left[\frac{2}{\mathrm{Re}}E^b\tilde{\dot{\boldsymbol{\varepsilon}}} - \tilde{\boldsymbol{\lambda}}\right] = \boldsymbol{e}_g.$$

As in the previous section, we will drop the tildes from now on.

Putting directly the transport term into the variational formulation would lead to the introduction of a trilinear form, which upon discretization would ultimately yield an asymmetric matrix A. Instead, we will use a timestepping algorithm and the first-order *characteristics* method presented in Section 5.2.2. The total derivative is discretized as in (5.15),

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u} = \frac{\mathbf{u}^{k+1} - \mathbf{u}^k\circ X^k}{\Delta_t} + O(\Delta_t), \qquad X^k(\boldsymbol{x}) := \boldsymbol{x} - \Delta_t\mathbf{u}^k(\boldsymbol{x}, t^k).$$

The variational formulation for each timestep is then obtained by taking the forms defined in Section 6.2.2 and adding the following terms to l and a,

$$\begin{aligned} a(\mathbf{u},\mathbf{v}) &:= a_{(6.2.2)}(\mathbf{u},\mathbf{v}) + \frac{1}{\Delta_t}\int_\Omega \langle \mathbf{u},\mathbf{v}\rangle \\ l(\mathbf{v}) &:= l_{(6.2.2)}(\mathbf{v}) + \frac{1}{\Delta_t}\int_\Omega \langle \mathbf{u}^k\circ X^k,\mathbf{v}\rangle. \end{aligned}$$

Solving for each timestep in the inertial setting is therefore equivalent to solving the problem defined in Section 6.2.

Remark 6.3. *Note that the timestepping scheme ensures a positive-definite form "a" even when $\mathrm{Re} = +\infty$. This allows us to extend the existence results of Property 6.4 to purely plastic flows, instead of being restricted to viscoplastic ones as in Section 6.2.*

Results We simulated the 2D silo shown in Figure 6.10, with $W = H = 8L$ and an aperture size D. We made the ratio $\tilde{D} := \frac{D}{L}$ vary from $\frac{1}{2}$ to 2, and studied the change in the dimensionless discharge rate $\tilde{Q}$ defined such that $Q := \sqrt{g}L^{\frac{3}{2}}\tilde{Q}$.

Plots for different rheologies, with the corresponding Beverloo fits when one was found, are shown in Figure 6.11. Beverloo law coefficients for $\mu = 0.6$, for which the fit was always acceptable except for $\mathrm{Re} = 1$, are given in Table 6.1. Coefficients for $\mu = 0.4$ are also given when the fit was deemed of sufficient quality. We remark that decreasing Re, increasing μ or increasing Bi all contribute to a consequent increase of k. μ and Bi have also a positive impact on C, while Re has only a small influence on this parameter.

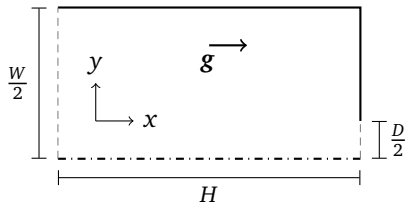


Figure 6.10: *Geometry of a half-silo of height H, width W and aperture size D*

		Re = 100		Re = 1000	Re = 10
		Bi = 0	Bi = 0.1	Bi = 0	
μ	C	1.70	1.44	1.77	2.00
	k	0	0.96	0	3.34
$\mu = 0.4$	C	1.54	1.52	1.54	1.48
	k	0.87	2.49	0.38	3.07
$\mu = 0.5$	C	1.35	1.43	1.35	1.33
	k	0.83	3.20	0.22	3.56

Table 6.1: *Beverloo law coefficents obtained for $\mu = 0.6$ and $\mu = 0.4$, for different Bi and Re values. The values for k are given assuming $L = 11.2D_g$.*

Extension to the $\mu(I)$ rheology The $\mu(I)$ (see Section 0.3.3) rheology is classically integrated into dynamics solvers by explicitly evaluating the value of the equivalent viscosity η_{eff} at each timestep (Lagrée et al. 2011).

In our case, we only have to explicitly evaluate the friction coefficient $\mu(I)$ instead of η_{eff}. Since $\mu(I)$ can only take values in $[\mu_S, \mu_D]$ whereas η_{eff} takes values in $\mathbb{R}^+$, our approach has a few benefits:

- We do not have to clamp the value of η_{eff}, and can have fully rigid zones where the shear rate is strictly zero (which means an infinite η_{eff});
- The loss of stability of the time-integration scheme due to this explicit term is much less dramatic;
- While fully implicit approaches are possible (Ionescu et al. 2015), our explicit update rule remains much simpler — and cheaper.

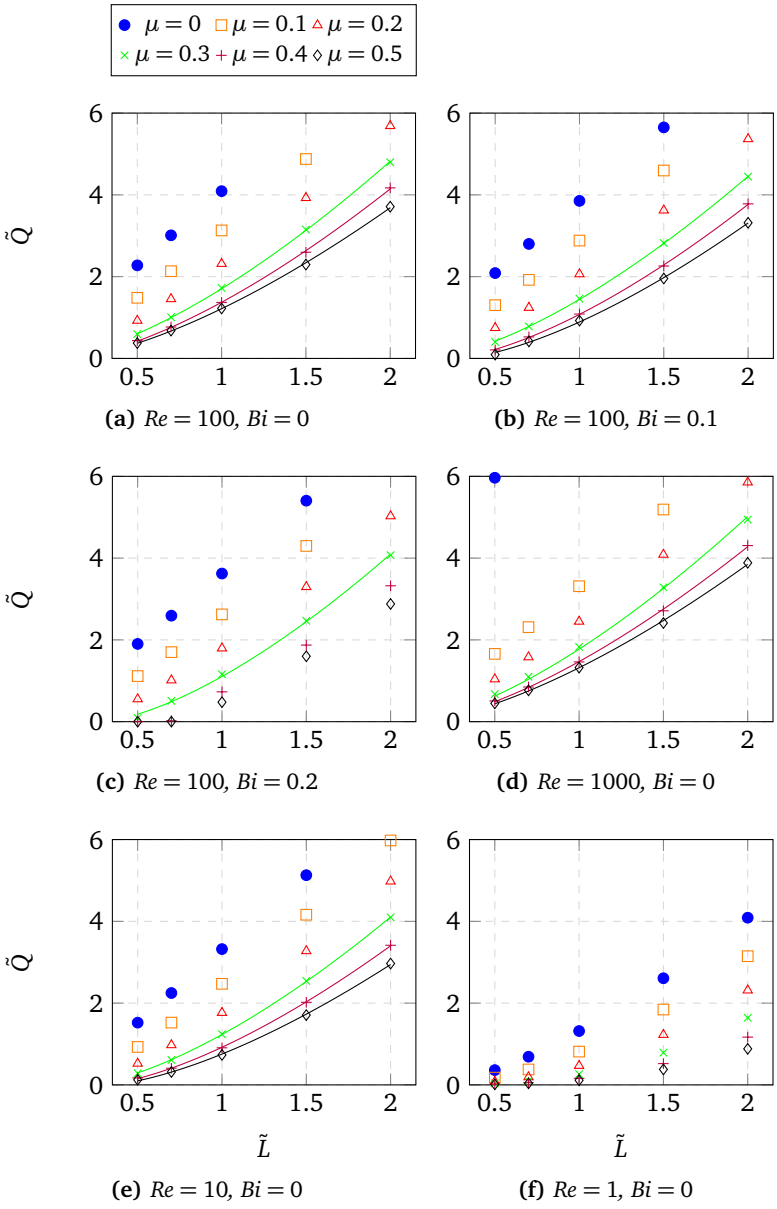
While the simulation frameworks are quite different — we do not take into account the air phase and use a non-zero (Re = 100) Newtonian viscosity, we nevertheless attempted to recreate a simulation from (Staron et al. 2014, Fig 4), using the same physical parameters ($D = 11.2D_g$, $I_0 = 0.4$, $\mu_S = 0.32$, $\mu_D = 0.6$).

The results are shown in Figure 6.12 ; the match on the C coefficient of the Beverloo law is surprisingly good (both methods give $C = 1.48$), however we retrieve a coefficient k that is significantly smaller (0.52 vs 0.73). Velocity profiles along the vertical and horizontal sections described in (Staron et al. 2014) are also presented in Figure 6.12.

6.5.4 Performance

We now study the computational performance for the problem of Section 6.2 – the dynamics case is just a sequence of such problems. In the following, **PG** will denote the canonical Projected-Gradient algorithm, **APGD** the Accelerated Projected Gradient Descent from Heyn (2013) — which is roughly similar to the FISTA* algorithm advocated in(Treskatis et al. 2016), **ASPG** the Accelerated Spectral Projected Gradient (Algorithm B.2), and **IP** the Interior-Point algorithm from the commercial package MOSEK (E. Andersen, Roos, et al. 2003).

In order to make as fair a comparison as possible, to evaluate the current error at each fixed-point iteration we have chosen not to use a residual based on a projection operator — which

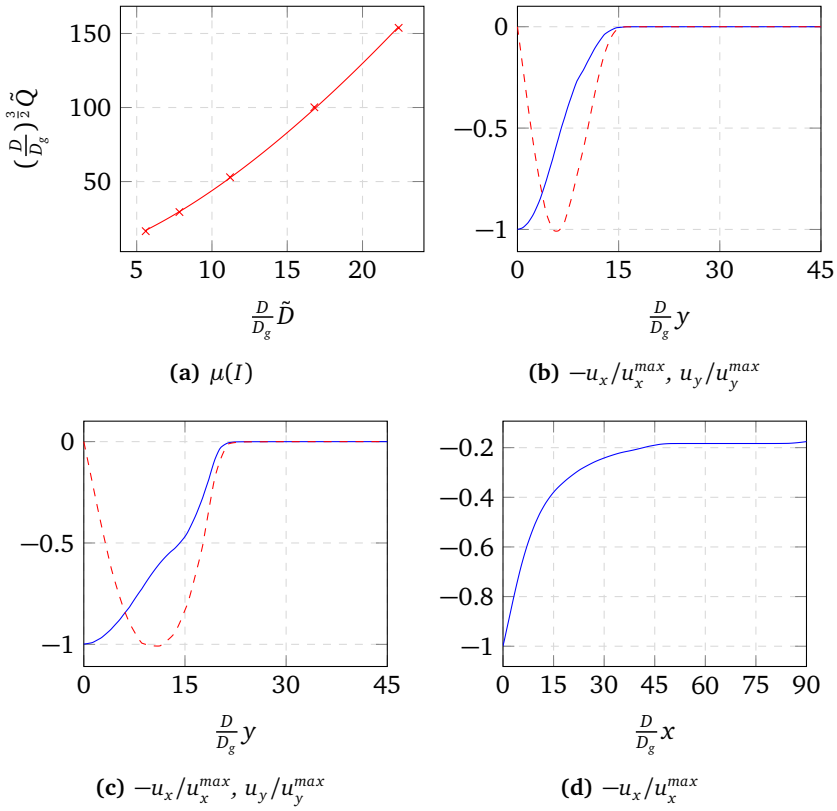


(a) $Re = 100$, $Bi = 0$ **(b)** $Re = 100$, $Bi = 0.1$

(c) $Re = 100$, $Bi = 0.2$ **(d)** $Re = 1000$, $Bi = 0$

(e) $Re = 10$, $Bi = 0$ **(f)** $Re = 1$, $Bi = 0$

Figure 6.11: *Dimensionless discharge rate $\tilde{Q}$ versus silo outlet diameter $\tilde{L}$ for and different rheologies. When μ is small, we cannot get a reasonable fit; in the case of the Newtonian flow, the discharge rate is closer to a linear law. Bi mainly influences the k parameter of the law but does suffices to obtain a fit. Large values of Re tend to linearize the discharge rate for small μ, smaller values make it quadratic.*



(a) $\mu(I)$ **(b)** $-u_x/u_x^{max}$, u_y/u_y^{max}

(c) $-u_x/u_x^{max}$, u_y/u_y^{max} **(d)** $-u_x/u_x^{max}$

Figure 6.12: *Top-left: Discharge rate as a function of the outlet size for the $\mu(I)$ rheology. The coefficients of the Beverloo fit are, using the dimensionless unit of (Staron et al. 2014), $C = 1.47$ and $k = 0.063$. Top-right and bottom: Horizontal and vertical velocity profiles for $\tilde{D} = 1$ along section $S1$, $S2$ and $H1$, as defined in (Staron et al. 2014, Fig. 6)*

	Mesh 1			**Mesh 2**			**Mesh 3**			**Mesh 4**		
Solver	t^1	n^2	t/n	t	n	t/n	t	n	t/n	t	n	t/n
IP	71	100	0.71	116	29	4	564	23	24	6281	17	369
ASPG	13	88	0.14	69	79	0.88	144	33	4.4	1098	47	23

[1] t : Total wall time in seconds.
[2] n : Number of iterations of the fixed-point algorithm.

Table 6.2: *Comparison of time taken to reach a give tolerance for 4 meshes with respectively 670, 2703, 10403 and 41509 vertices.*

are known to be disadvantageous for interior-points — but rather the SOC Fischer-Burmeister function for frictional problems as defined e.g. in Chapter 4.

We consider the silo geometry of Figure 6.10, a $\mathbb{P}_2 - \mathbb{P}_{1d}$ FEM approximation order, and a rheology with $\mu = 0.5$, $\mathrm{Re} = 1$ and $\mathrm{Bi} = 0$. The base mesh has a resolution of 640 vertices.

Figure 6.13 shows typical convergence plots for selected algorithms. Table 6.2 shows the evolution of the computation time in seconds as the mesh is refined, for an interior-point method and a Spectral Projected Gradient method. The latter, despite its simplicity, is therefore highly competitive.

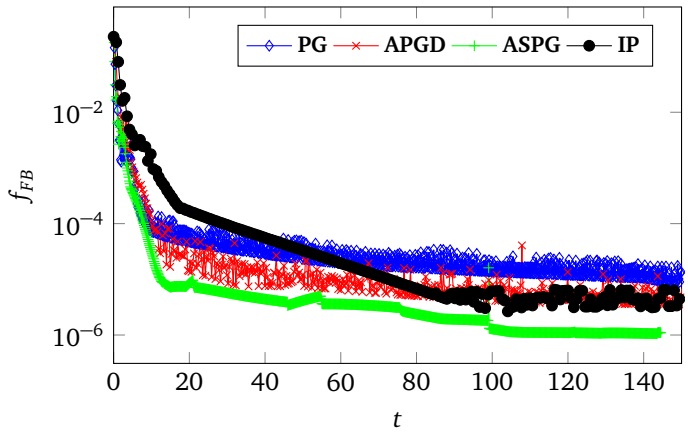


Figure 6.13: *Infinity-norm of the Fischer-Burmeister function as a function of the wall time for the Projected-Gradient, Accelerated Projected Gradient Descent, Spectral Projected Gradient and Interior-Point algorithm*

Discussion

In this chapter we have presented a macroscopic, continuous model for granular-like flows, and we have shown that it was driven by equations structurally similar to those appearing in contact mechanics between discrete elements. We have then exhibited algorithms for solving those equations on a class of spatial discretization and demonstrated that they were well-defined.

These new numerical simulation tools allowed us to capture the typical nonsmooth effects critical to the simulation of yield-stress flows, as well as to accommodate complex rheologies such as the well-known $\mu(I)$ rheology. The main advantages over most previous approaches are the absence of any regularization or clamping, and improved efficiency over Augmented Lagrangian algorithms, thanks to formulations in the framework of Second Order Cone Programming.

Limitations The computational cost of our approach remains a major limitation. The SOCP formulation becomes quickly expensive when the number of required fixed-point iterations increases, while the Projected Gradient requires solving at least one linear system at each inner iteration. This makes three-dimensional scenarios only tractable for rough meshes. However, note that for the timestepping algorithm without Newtonian viscosity ($\mathrm{Re} = +\infty$), the inverse of the A matrix will be sparse for certain discretizations of $V(\Omega)$. The explicit computation of the Delassus operator would then become possible. In the next chapter, we will propose an approximation for large Re that makes use of this fact. This will both speed up our numerical method and allow us to use other DCFP solvers, such as our hybrid Gauss–Seidel algorithm from Chapter 4.

From a physics point of view, the fact that we do not take into account a variable density leads to invalid predictions in zones where the local divergence is strictly positive, such as regions in the wake of obstacles — or even any shearing region in the case of a non-zero dilatancy coefficient. Our simple unilateral compressibility condition $\nabla \cdot \mathbf{u} \geq 0$ also prevents the material from recompacting after dilatation; a condition based on the current volume fraction, such as proposed by Dunatunga and Kamrin (2015), should be used instead. The next chapter will be dedicated to the treatment of a proper maximal volume fraction constraint, thus extending our approach to general scenarios. Our treatment of boundary conditions is also limited. As argued in (Domnik and Pudasaini 2012), we should consider more realistic laws, such as Coulombic ones.

Finally, as usual when dealing with Coulomb-like friction models, the theoretical existence and uniqueness criterions for our solutions are rather weak. Not being able to uniquely de-

termine the stress field for a given velocity solution is problematic for assessing the stress on structures. However, as the flowing and dilating regions enforce boundary conditions for the stress inside the rigid zones, we did not find this under-determination to be problematic for our test scenarios.

7 Dry granular flows

In this chapter, we relax the dense flow hypothesis from the framework presented in Chapter 6. As such, we shall be able to simulate arbitrary compacting and dilating granular flows. Note that a large part of the contents of this chapter has been published in (Daviet and Bertails-Descoubes 2016a).

Just like in (Narain, Golas, et al. 2010) and (Dunatunga and Kamrin 2015), we shall assume that the volume fraction of grains ϕ fully determines the nature of the contacts between the grains. That is, we assume the existence of a critical value $0 < \phi_{\max} \leq 1$ such that for $\phi < \phi_{\max}$, we consider that the grains are mostly separated, and that they are interacting though sparse random impacts; the material is in the gaseous regime. However, for $\phi = \phi_{\max}$, we shall consider that the grains are in permanent contact, and that their interactions are driven by dry Coulomb-like friction. The material is then either in the liquid or the solid regime, and subject to the non-associated Drucker–Prager flow rule $\mathscr{DP}(\mu, \sigma_S, \tau_c, \zeta)$. In practice, we used $\phi_{\max} = 0.6$, corresponding to a loose 3D monodisperse packing.

As in the previous chapter, we shall also assume that the material is inelastic (the grains are fully rigid, and thus the volume fraction ϕ should not exceed the critical value $\phi_{\max}$), and that the interactions between the grains and the surrounding fluid can be neglected. Moreover, for the sake of simplicity we shall no longer consider non-zero values for the shear yield stress σ_S and the dilatancy ζ. Following the considerations of Section 6.4, the adaptation of the numerical solvers to handle those cases will be straightforward.

7.1 Spatially continuous model

7.1.1 Constitutive equations

As we neglect the effect of the surrounding air, the total stress of the grain–air mixture with volume fraction of grains ϕ can be expressed as $\boldsymbol{\sigma} = \phi \boldsymbol{\sigma}^g$, where $\boldsymbol{\sigma}^g$ is the granular phase stress. As in the previous chapter, we decompose the solid phase stress tensor as $\boldsymbol{\sigma}^g := \eta \dot{\boldsymbol{\epsilon}} + \boldsymbol{\sigma}^C$, where the first part corresponds to a standard Newtonian viscosity (dissipative term due to random collisions in the flowing material), and $\boldsymbol{\sigma}^C$ is the additional stress due to the Coulombic interactions between individual grains. Note that η may be chosen equal to zero, in this case no internal stress is applied in the "gaseous" phase and we retrieve the constitutive law used by Narain, Golas, et al. (2010). The normal contact stress (a.k.a. "pressure") is $p = -\frac{1}{d} \operatorname{Tr} \boldsymbol{\sigma}^C$, such that $\boldsymbol{\sigma}^C = \operatorname{Dev} \boldsymbol{\sigma}^C - p\mathbb{I}$.

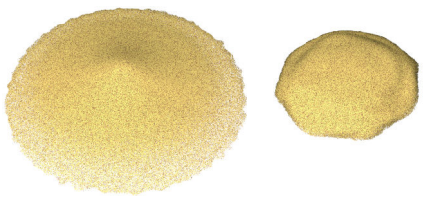


Figure 7.1: *Final states after the collapse of a cylindrical column with exact (Frobenius norm, left) and linearized (ℓ_∞-norm, right) Drucker-Prager yield surfaces, both for $\mu = 0.6$.*

Just like in the previous chapter, we consider a Drucker–Prager yield criterion with friction coefficient $\hat{\mu}$ and cohesion c, that is, a pressure-dependent yield stress $\hat{\mu}(p+c)$. However, following Narain, Golas, et al. (2010), the pressure no longer enforces a positive divergence of the flow, but the maximal volume fraction constraint $\phi \leq \phi_{\max}$. That is, we replace Equation (6.3) with the complementarity relationship

$$0 \leq p + c \perp \phi_{\max} - \phi \geq 0. \tag{7.1}$$

The maximum dissipation principle states that in the yielded regime, friction should be saturated and the frictional stress tensor should be colinear to the deviatoric part of the strain rate. The deviatoric part of $\boldsymbol{\sigma}^C$ should thus satisfy one of the two regimes,

$$\begin{cases} \operatorname{Dev}\boldsymbol{\sigma}^C = \hat{\mu}\,(p+c)\,\dfrac{\operatorname{Dev}\dot{\boldsymbol{\varepsilon}}}{|\operatorname{Dev}\dot{\boldsymbol{\varepsilon}}|} & \text{if } \operatorname{Dev}\dot{\boldsymbol{\varepsilon}} \neq \mathbf{0} \text{ (yielded)} \\ |\operatorname{Dev}\boldsymbol{\sigma}^C| \leq \hat{\mu}(p+c) & \text{if } \operatorname{Dev}\dot{\boldsymbol{\varepsilon}} = \mathbf{0} \text{ (unyielded).} \end{cases} \tag{7.2}$$

Note that in contrast with (Narain, Golas, et al. 2010), we will not linearize the $|\cdot|$ norm in Equation (7.2), thus avoiding the disturbing geometrical artifacts inherent to their method (Figure 7.1).

Remark 7.1. *When $\phi < \phi_{max}$, Equation (7.1) imposes $p+c=0$, and thus Equation (7.2) dictates that the frictional stress should vanish, which is expected. However, the mixture pressure will be $-\phi c$ in the gaseous zones; if $c>0$, the grains will thus tend to attract each other even though the contacts are broken, which is unwanted. For this reason, the cohesion field c should also satisfy $\phi < \phi_{max} \implies c = 0$. This may create spatial discretization issues, as we will see later.*

7.1.2 Energy considerations

Let us study the conditions for which the flow rule defined by Equation (7.1–7.2) and the conservation equations (5.7,5.10) lead to a dissipative system, as is expected from a granular material. We have already performed a similar study in the supplemental document to (Daviet and Bertails-Descoubes 2016a) for the discrete-time version of the rheology. We propose here to look at the continuous-time setting instead, by leveraging an analogy with impacts in discrete contact mechanics.

Local dissipativity of contact stress Let us consider a material point $\boldsymbol{x}(t)$ of the domain, with $\phi(t) := \phi(\boldsymbol{x}(t), t)$. We will see that both the inelastic impacts hypothesis and the criterion from Remark 7.1, that cohesion should vanish in non-dense zones, are critical to ensure that the work of the contact stress is always and everywhere dissipative.

The density of energy dissipation by the contact stress is given by $(-\phi(t)\boldsymbol{\sigma}^C(t)) : \dot{\boldsymbol{\varepsilon}}(t)$. For the system to be locally dissipative, it suffices that $\phi\boldsymbol{\sigma}^C : \dot{\boldsymbol{\varepsilon}} \geq 0$, or, *a fortiori*, that

$$\begin{cases} \operatorname{Dev}\boldsymbol{\sigma}^C : \operatorname{Dev}(\phi\,\mathrm{D}(\mathbf{u})) \geq 0 & \text{(7.3a)} \\ \operatorname{Tr}\boldsymbol{\sigma}^C \operatorname{Tr}(\phi\,\mathrm{D}(\mathbf{u})) \geq 0. & \text{(7.3b)} \end{cases}$$

The implication (7.1–7.2) $\implies$ (7.3a) is trivial; either $\operatorname{Dev}(\phi\,\mathrm{D}(\mathbf{u})) = 0$, or $\operatorname{Dev}(\phi\,\mathrm{D}(\mathbf{u})) \neq 0$ and $\operatorname{Dev}\boldsymbol{\sigma}^C = \hat{\mu}\,(p+c)\,\frac{\operatorname{Dev}\boldsymbol{\gamma}}{|\operatorname{Dev}\boldsymbol{\gamma}|}$ with $p+c \geq 0$.

Now, (7.3b) can be written equivalently as $(\mathrm{p}\,\phi\,\nabla\cdot\mathbf{u}) \leq 0$, or again, using the mass conservation equation (6.1), $\mathrm{p}\frac{D\phi}{Dt} \geq 0$. Using an analogy with discrete mechanics, we argue below that this inequality shall hold if the consideration made in Remark 7.1 is satisfied.

The maximal volume fraction constraint, $\phi_{\max} - \phi(t) \geq 0$, can be seen as analogous to the normal part of the *gap* function in contact mechanics. The instant at which ϕ reaches $\phi_{\max}$ constitutes an impact, and the time-derivative of the gap function (the normal relative velocity

for discrete contacts, $\frac{D\phi}{Dt}(\boldsymbol{x})$ here) will be discontinuous. However, this derivative will retain both a left and a right limit at every instant.

Pursuing the analogy with nonsmooth discrete mechanics, we interpret our inelastic impacts hypothesis as preventing rebounds; that is, if ϕ reach $\phi_{\max}$ at time t, then we shall have $\frac{D\phi}{Dt}(\boldsymbol{x}(t), t^+) = 0$. Moreover, just like we did to construct the Signorini conditions, we assume that shocks do not propagate. Therefore, if t is not a time of impact for the material point $\boldsymbol{x}(t)$, then the contact pressure $p(t) := \mathrm{p}(\boldsymbol{x}(t), t)$, should possess both left and right limits.

The volume fraction and its derivative have thus to satisfy one of the following four cases,

$$\begin{cases} \phi(t) < \phi_{\max} & \text{(7.4a)} \\ \text{or } \phi(t) = \phi_{\max} \text{ and } \dfrac{D\phi}{Dt}(\boldsymbol{x}(t), t^-) > 0 \text{ and } \dfrac{D\phi}{Dt}(\boldsymbol{x}(t), t^+) = 0 & \text{(7.4b)} \\ \text{or } \phi(t) = \phi_{\max} \text{ and } \dfrac{D\phi}{Dt}(\boldsymbol{x}(t), t) = 0 & \text{(7.4c)} \\ \text{or } \phi(t) = \phi_{\max} \text{ and } \dfrac{D\phi}{Dt}(\boldsymbol{x}(t), t^-) > 0 \text{ and } \dfrac{D\phi}{Dt}(\boldsymbol{x}(t), t^+) = 0. & \text{(7.4d)} \end{cases}$$

Case (7.4a) corresponds to the gaseous regime, case (7.4b) to an impact, case (7.4c) to persistent contact and case (7.4d) to the onset of dilation.

Let us show that, under the hypothesis of Remark 7.1, i.e., that $\phi < \phi_{\max} \implies c = 0$, then $\mathrm{p}\frac{D\phi}{Dt} \geq 0$. This is obvious when $\frac{D\phi}{Dt} = 0$, let us consider the remaining cases.

- Case (7.4a). The complementarity condition (7.1) requires $p = -c = 0$.
- Case (7.4b) at t^-. Left-continuity of $\frac{D\phi}{Dt}$ means that there exists t_0 such that $\forall s \in [t_0, t[$, $\frac{D\phi}{Dt}(s) < 0$, and thus $\phi(s) < \phi_{\max}$, meaning $c(s) = 0$. Left-continuity of c imposes in turn that $c(t^-) = 0$, and condition (7.1) that $p(t^-) \geq 0$. We deduce $(\mathrm{p}\frac{D\phi}{Dt})(\boldsymbol{x}, t^-) \geq 0$.
- Case (7.4d) at t^+. As t is not a time of impact, by analogy with discrete mechanics we assume right-continuity of p. There exists t_0 such that $\forall s \in]t, t_0]$, $\phi(s) < \phi_{\max}$, thus $c(s) = p(s) = 0$, and $p(t^+) = 0$. Once again $(\mathrm{p}\frac{D\phi}{Dt})(\boldsymbol{x}(t), t^+) = 0$.

Global dissipativity For the sake of simplicity, we shall consider a domain Ω with homogeneous Dirichlet boundary conditions.

We can decompose the total energy of the system as the sum of its potential and kinetic energies, i.e., $\mathscr{E}^t = \mathscr{E}^p + \mathscr{E}^c$, with

$$\mathscr{E}^p := -\int_\Omega \rho\phi \langle \mathbf{x}, \boldsymbol{g} \rangle \qquad \mathscr{E}^c := \int_\Omega \frac{1}{2}\rho\phi \langle \mathbf{u}, \mathbf{u} \rangle .$$

Since we are using homogeneous Dirichlet boundary conditions, their evolution follows

$$\begin{aligned} \frac{\mathrm{d}\mathscr{E}^p}{\mathrm{d}t} &= -\rho \int_\Omega \frac{\partial \phi}{\partial t} \langle \mathbf{x}, \boldsymbol{g} \rangle = \rho \int_\Omega \nabla \cdot [\phi \mathbf{u}] \langle \mathbf{x}, \boldsymbol{g} \rangle \\ &= -\rho \int_\Omega \phi \langle \mathbf{u}, \mathbf{g} \rangle + \rho \underbrace{\int_{\mathrm{Bd}\,\Omega} \phi \langle \mathbf{u}, \boldsymbol{n}_\Omega \rangle \langle \boldsymbol{x}, \mathbf{g} \rangle}_{=0}, \end{aligned}$$

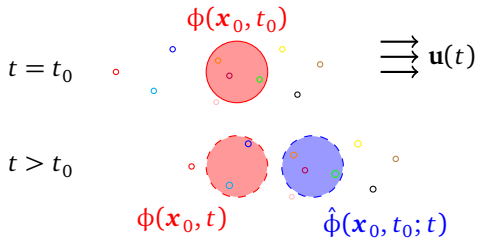


Figure 7.2: *Eulerian (in red) and Lagrangian (in blue) points of view for studying the change in volume fraction of a compressible fluid.*

and

$$\begin{aligned}\frac{\mathrm{d}\mathscr{E}^c}{\mathrm{d}t} &= \int_\Omega \rho\phi \left\langle \frac{D\mathbf{u}}{Dt}, \mathbf{u} \right\rangle \\ &= \int_\Omega \eta \left\langle \nabla\cdot[\phi\dot{\boldsymbol{\varepsilon}}], \mathbf{u}\right\rangle + \left\langle \nabla\cdot\left[\phi\boldsymbol{\sigma}^C\right], \mathbf{u}\right\rangle + \rho\phi\left\langle \boldsymbol{g}, \mathbf{u}\right\rangle \\ &= \rho\int_\Omega \phi\left\langle \mathbf{u}, \mathbf{g}\right\rangle - \eta\underbrace{\int_\Omega \phi\dot{\boldsymbol{\varepsilon}} : \dot{\boldsymbol{\varepsilon}}}_{\geq 0} - \int_\Omega \phi\boldsymbol{\sigma}^C : \dot{\boldsymbol{\varepsilon}} \\ &\quad + \underbrace{\int_{\mathrm{Bd}\,\Omega} \phi\left\langle \mathbf{u}, \left(\eta\dot{\boldsymbol{\varepsilon}} + \boldsymbol{\sigma}^C\right)\boldsymbol{n}_\Omega\right\rangle}_{=0}.\end{aligned}$$

Therefore $\frac{\mathrm{d}\mathscr{E}^t}{\mathrm{d}t} \leq -\int_\Omega \phi\boldsymbol{\sigma}^C : \dot{\boldsymbol{\varepsilon}}$. Provided the local dissipativity of the contact stress that we discussed in the previous paragraph, the overall material is thus also dissipative.

7.1.3 Semi-implicit integration

Unlike in Chapter 6, our rheology involves the volume fraction field ϕ; the mass conservation and momentum balance equations are now fully coupled. Moreover, as we are using an inelastic model — i.e., in the infinite compression Young modulus limit — numerical stability considerations impose using an integration scheme with some degree of implicity.

We propose using a two-step semi-implicit algorithm. At the k^{th} timestep,

1. we first solve the momentum balance equation (5.10) using the current volume fraction field ϕ^k, and deduce the end-of-step velocity and stress fields $\mathbf{u}^{k+1}$ and $\boldsymbol{\sigma}^{k+1}$;
2. we then solve the mass conservation equation (5.7) using the end-of-step velocities $\mathbf{u}^{k+1}$ and deduce ϕ^{k+1}.

However, note that Step 1. requires finding $\mathbf{u}^{k+1}$ and $\boldsymbol{\sigma}^{k+1}$ such that the end-of-step velocity constraints are satisfied, and in particular, the end-of-step maximum volume fraction constraint, $\phi^{k+1} \leq \phi_{\max}$. Yet ϕ^{k+1} is only computed at Step 2.! We must therefore replace the maximum volume fraction constraint in Step 1. with an approximation of ϕ^{k+1} that can be computed using only ϕ^k and $\mathbf{u}^{k+1}$, and is much cheaper to evaluate than solving the full mass conservation equation. The following section is dedicated to this choice of approximation.

Remark 7.2. *Our algorithm could be made fully implicit by keeping iterating between the two steps until a fixed point for the volume fraction and velocity fields is reached.*

Maximum volume fraction The mass conservation equation (5.7) describes how the volume fraction changes in time. Yet, as depicted in Figure 7.2, there are two ways to express this

change. On the one hand, the Eulerian point of view looks at the change of ϕ at a fixed location in space $\boldsymbol{x}_0$, which is given by the derivative $\frac{\partial \phi}{\partial t}$ evaluated at $\boldsymbol{x}_0$. We have

$$\begin{aligned}\phi(\boldsymbol{x}_0, t) &= \phi(\boldsymbol{x}_0, t_0) + \int_{t_0}^{t} \frac{\partial \phi}{\partial t}(\boldsymbol{x}_0, s)\mathrm{d}s \\ &= \phi(\boldsymbol{x}_0, t_0) - \int_{t_0}^{t} \nabla \cdot [\phi \mathbf{u}](\boldsymbol{x}_0, s)\mathrm{d}s, \qquad \text{using (5.7)}.\end{aligned}$$

On the other hand, the Lagrangian point of view follows grains as they move through space. The volume fraction at such a tracked point is given by $\hat{\phi}(\boldsymbol{x}_0, t_0; t) := \phi(X(\boldsymbol{x}_0, t_0; t), t)$, where $X(\boldsymbol{x}_0, t_0; t)$ is the *characteristics* function defined by the Cauchy problem (5.13). Using the chain derivation rule w.r.t. time (5.14) and the mass conservation equation (5.7), we get

$$\frac{\partial \hat{\phi}}{\partial t}(\boldsymbol{x}_0, t_0; t) = \frac{D\phi}{Dt}(X(\boldsymbol{x}_0, t_0; t), t) = -\phi \nabla \cdot \mathbf{u}(X(\boldsymbol{x}_0, t_0; t), t).$$

We thus have

$$\begin{aligned}\hat{\phi}(\boldsymbol{x}_0, t_0; t) &= \hat{\phi}(\boldsymbol{x}_0, t_0; t_0) + \int_{t_0}^{t} \frac{\partial \phi}{\partial t}(\boldsymbol{x}_0, t_0, s)\mathrm{d}s \\ &= \phi(\boldsymbol{x}_0, t_0) - \int_{t_0}^{t} (\phi \nabla \cdot \mathbf{u})(X(\boldsymbol{x}_0, t_0; s), s)\mathrm{d}s.\end{aligned} \tag{7.5}$$

The maximum volume fraction constraint, $\phi \leq \phi_{\max}$, can be expressed at every instant and every point of the simulation domain using either the Eulerian or Lagrangian point of views. Studying the rate of change eventually leads to conditions on $\nabla \cdot [\phi \mathbf{u}]$ or $\phi \nabla \cdot \mathbf{u}$, respectively. Both conditions are equivalent in the spatially continuous case, however they will yield different discretizations. A simple physical consideration will drive our choice: at the grain scale, the contact forces oppose the relative velocity of the particles; at the macroscopic scale, we can thus expect that the pressure will oppose the (opposite of) the divergence of the flow. Hence, we will use the Lagrangian point of view to discretize the maximum volume fraction constraint.

First-order approximation With the Lagrangian framework of Equation (7.5), we linearize the constraint $\hat{\phi}^{k+1} \leq \phi_{\max}$ on a time interval $[t^k, t^{k+1} := t^k + \Delta_t]$ as $\left(\phi^k - \Delta_t \phi^k \nabla \cdot \mathbf{u}^{k+1}\right) \leq \phi_{\max}$. Equation (7.1) can thus be approximated at the first order as

$$0 \leq \phi^k \nabla \cdot \mathbf{u}^{k+1} + \beta^k \perp \mathrm{p}^{k+1} + \mathrm{c}^k \geq 0, \tag{7.6}$$

where $\beta^k := \frac{\phi_{\max} - \phi^k}{\Delta_t}$ is a scalar field expressing the maximum rate at which the material can compress during Δ_t.

7.1.4 Discrete-time equations

We now perform a few mathematical manipulations that will reveal a problem structure similar to that of the previous chapter. To lighten the notations, we will omit the $k+1$ superscript; unless otherwise mentioned, the fields are evaluated at the end of the time-step.

As mentioned in Remark 7.1, the frictional contact stress field $\mathrm{Dev}\,\boldsymbol{\sigma}^C$ vanishes when the material is not at maximum compaction, and in particular where there is no granular matter ($\phi^k = 0$). One may thus rewrite (7.2) equivalently as

$$\begin{cases} \mathrm{Dev}\,\boldsymbol{\sigma}^C = \hat{\mu}\left(\mathrm{p} + \mathrm{c}^k\right) \dfrac{\phi^k \,\mathrm{Dev}\,\dot{\boldsymbol{\varepsilon}}}{|\phi^k \,\mathrm{Dev}\,\dot{\boldsymbol{\varepsilon}}|} & \text{if } \phi^k \,\mathrm{Dev}\,\dot{\boldsymbol{\varepsilon}} \neq \mathbf{0} \\ |\mathrm{Dev}\,\boldsymbol{\sigma}^C| \leq \hat{\mu}(\mathrm{p} + \mathrm{c}^k) & \text{if } \phi^k \,\mathrm{Dev}\,\dot{\boldsymbol{\varepsilon}} = \mathbf{0}. \end{cases} \tag{7.7}$$

We now consider the friction coefficient $\mu := \sqrt{\frac{2}{d}}\hat{\mu}$, and the symmetric tensor fields $\boldsymbol{\lambda} := c^k\mathbb{I} - \boldsymbol{\sigma}^C$ and $\boldsymbol{\gamma} := \phi\dot{\boldsymbol{\varepsilon}} + \frac{\beta}{d}\mathbb{I}$. We have $\phi^k\nabla\cdot\mathbf{u} + \beta = \operatorname{Tr}\boldsymbol{\gamma}$, $\phi^k \operatorname{Dev}\dot{\boldsymbol{\varepsilon}} = \operatorname{Dev}(\boldsymbol{\gamma})$, and $\mu\boldsymbol{\lambda}_N = \hat{\mu}\sqrt{\frac{2}{d}}\left(\sqrt{\frac{d}{2}}c^k - \frac{\operatorname{Tr}\boldsymbol{\sigma}^C}{\sqrt{2d}}\right) = \hat{\mu}(c^k + p)$, so that Equations (7.6)–(7.7) reread

$$\begin{cases} \operatorname{Dev}\boldsymbol{\lambda} = -\mu\boldsymbol{\lambda}_N\dfrac{\operatorname{Dev}\boldsymbol{\gamma}}{|\operatorname{Dev}\boldsymbol{\gamma}|} & \text{if } \operatorname{Dev}\boldsymbol{\gamma} \neq 0 \\ |\operatorname{Dev}\boldsymbol{\lambda}| \leq \mu\boldsymbol{\lambda}_N & \text{if } \operatorname{Dev}\boldsymbol{\gamma} = 0 \\ \qquad 0 \leq \boldsymbol{\gamma}_N \perp \boldsymbol{\lambda}_N \geq 0. \end{cases} \tag{7.8}$$

We recognize the disjunctive formulation (1.20) of the non-associated Drucker–Prager flow rule expressed on $\boldsymbol{\gamma}$ and $\boldsymbol{\lambda}$,

$$\boldsymbol{\gamma} \text{ and } \boldsymbol{\lambda} \text{ satisfy (7.8)} \iff (\boldsymbol{\gamma}, \boldsymbol{\lambda}) \in \mathscr{DP}(\mu).$$

Conservation of momentum Replacing the expression of $\boldsymbol{\lambda}$ into the momentum conservation equation (5.10), the discrete time momentum balance equation that we will have to solve at Step 1. of our semi-implicit integration algorithm reads

$$\rho\phi^k\frac{D\mathbf{u}}{Dt} - \nabla\cdot\left[\eta\phi^k \operatorname{D}(\mathbf{u}) - \phi^k\boldsymbol{\lambda}\right] = \rho\phi^k\boldsymbol{g} + \nabla\left[\phi^k c^k\right]. \tag{7.9}$$

The total derivative remains to be discretized, but the method for doing so will be tightly linked to our choice of spatial discretization. For now, we will assume that $\frac{D\mathbf{u}}{Dt}$ can be approximated at the first order as

$$\frac{D\mathbf{u}}{Dt} = \frac{1}{\Delta_t}\left(\mathbf{u} - u(\mathbf{u}^k)\right) + O(\Delta_t),$$

with the mapping u to be defined later.

Variational formulation Similarly as in Chapter 6, the momentum conservation equation (7.9) with the non-associated Drucker-Prager flow rule can be put under variational form. Let V denote the subspace of $H^1(\Omega)^d$ satisfying the homogeneous Dirichlet boundary conditions, and let $T \subset L_2(\Omega)^s_d$ a subspace of square-integrable symmetric tensor fields on Ω.

Proposition 7.1. *The first step of out semi-implicit integration scheme amounts to finding $\mathbf{u} \in V$, $\boldsymbol{\gamma}, \boldsymbol{\lambda} \in T^2$ such that*

$$\begin{cases} m(\mathbf{u},\mathbf{v}) + a(\mathbf{u},\mathbf{v}) = b(\boldsymbol{\lambda},\mathbf{v}) + l(\mathbf{v}) & \forall \mathbf{v} \in T \\ s(\boldsymbol{\gamma},\boldsymbol{\tau}) = b(\boldsymbol{\tau},\mathbf{u}) + k(\boldsymbol{\tau}) & \forall \boldsymbol{\tau} \in T \\ (\boldsymbol{\gamma},\boldsymbol{\tau}) \in \mathscr{DP}(\mu), \end{cases} \tag{7.10}$$

where the bilinear forms a, b, m and linear forms l,k are given by

$$m(\mathbf{u},\mathbf{v}) := \frac{\rho}{\Delta_t}\int_\Omega \phi^k\langle\mathbf{u},\mathbf{v}\rangle \qquad s(\boldsymbol{\gamma},\boldsymbol{\tau}) := \int_\Omega \boldsymbol{\gamma}:\boldsymbol{\tau}$$

$$a(\mathbf{u},\mathbf{v}) := \eta\int_\Omega \phi^k \operatorname{D}(\mathbf{u}):\operatorname{D}(\mathbf{v}) \qquad b(\boldsymbol{\tau},\mathbf{u}) := \int_\Omega \phi^k\boldsymbol{\tau}:\operatorname{D}(\mathbf{u})$$

$$l(\mathbf{v}) := \rho\int_\Omega \phi^k\left(\boldsymbol{g} + \frac{u(\mathbf{u}^k)}{\Delta_t}\right) \qquad k(\boldsymbol{\tau}) := \int_\Omega \frac{\max\left(\phi_{max} - \phi^k, 0\right)}{\Delta_t}\frac{\operatorname{Tr}\boldsymbol{\tau}}{d}.$$

Proof. One can see easily that the second line of 7.10 is just the definition of $\boldsymbol{\gamma}$ put under weak form. Multiplying the conservation of momentum (7.9) by a test function $\mathbf{v}$, one gets

$$\frac{\rho}{\Delta_t}\int_\Omega \phi^k\langle\mathbf{u},\mathbf{v}\rangle - \eta\int_\Omega\left\langle\nabla\cdot\left[\phi^k \operatorname{D}(\mathbf{u})\right],\mathbf{v}\right\rangle = -\int_\Omega\left\langle\nabla\cdot\left[\phi^k\boldsymbol{\lambda}\right],\mathbf{v}\right\rangle + \rho\int_\Omega \phi^k\left(\boldsymbol{g} + \frac{u(\mathbf{u}^k)}{\Delta_t}\right).$$

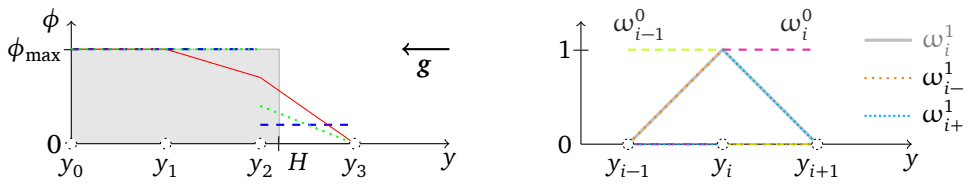


Figure 7.3: ***(a)*** *Fraction field for a volume ($H\phi_{max}$) of grains at rest under gravity; exact (gray), and using different approximations: piecewise-linear continuous ($\mathbb{P}_1$, line), piecewise-constant ($\mathbb{P}_0$, dashed), piecewise-linear discontinuous ($\mathbb{P}_{1d}$, dotted).* ***(b)*** *Shape functions for those low-order approximations.*

Then, using the Green formula and taking into account that the test function $\mathbf{v}$ vanishes on the boundary of Ω,

$$b(\boldsymbol{\lambda},\mathbf{v}) = \int_\Omega \phi^k \boldsymbol{\lambda} : \mathrm{D}(\mathbf{v}) = -\int_\Omega \left\langle \nabla\cdot\left[\phi^k\boldsymbol{\lambda}\right],\mathbf{v}\right\rangle$$
$$a(\mathbf{u},\mathbf{v}) = -\eta\int_\Omega \left\langle \nabla\cdot\left[\phi^k\,\mathrm{D}(\mathbf{u})\right],\mathbf{v}\right\rangle.$$

□

We notice that the variational formulation (7.10) is structurally similar to the one governing the dense case, System (6.13a–6.13c). From there, it would seem straightforward to use once again the discretization strategy discussed in Chapter 6. However, we will see in the next section that supplemental restrictions on the choice of elements apply.

Note also the "max" function in the expression of the bilinear form k, which at first could appear superfluous. However, it may happen that during the simulation, ϕ locally grows above $\phi_{\max}$. Without the "max" clamping, the following timestep will attempt to correct the local volume fraction excess, and doing so may introduce energy and thus visual popping artifacts. Indeed, it can be easily verified that the dissipativity property that we exhibited in the time-continuous setting (Section 7.1.2) will remain valid in the discrete case only if β^k, defined such that $\boldsymbol{\gamma} = \phi\dot{\boldsymbol{\varepsilon}} + \beta^k\frac{\mathbb{I}}{d}$, is positive. This is what we ensure here by setting $\beta^k = \frac{1}{\Delta_t}\max\left(\phi_{\max}-\phi^k,0\right)$ instead of its original definition as per Equation (7.6). The main drawback of this approach is that by not attempting to correct volume fraction excesses, the simulation may visually lose volume over time.

7.2 Discretization using finite elements

7.2.1 Space compability criterion

A simple compability consideration will greatly reduce the choice of finite-element spaces that we can use to discretize our problem. Let us consider the $1D$ case of a cohesionless, frictionless material. We want to be able to express the equilibrium of the material under gravity; this reads

$$\nabla[\phi \mathrm{p}] = -\phi\rho g. \tag{7.11}$$

Moreover, the Drucker–Prager rheology imposes that $\phi < \phi^{\max} \implies \mathrm{p} = 0$. Necessarily, the equilibrium (7.11) requires that every point for which $\phi > 0$ is within the stencil of the gradient operator evaluated at a point for which $\phi = \phi_{\max}$. In other words, ϕ should decrease "abruptly enough". Note that this is the case of the physical solution: at equilibrium, there exists H such that $\phi(x) = \phi_{\max}$ for $x \le H$, and $\phi(x) = 0$ for $x > H$.

More formally, let V_h and T_h be subspaces of $V \subset H^1(\mathbb{R})$ and $T \subset L_2(\mathbb{R})$. The complementarity condition (7.1) suggests using the same discretization space for the volume fraction and stress fields, so we will assume $\phi \in T$. Consider the linear form l_h on V_h, $l_h(\mathbf{v}_h) := \int_\Omega \phi_h \mathbf{v}_h$, and the bilinear form on $T_h \times V_h$ $b_h(\mathrm{p}_h, \mathbf{v}_h) := \int_\Omega \nabla[\mathrm{p}_h]\mathbf{v}_h$. Let $\Lambda(\phi_h) \subset T_h$ be the linear subspace of the pressure fields that are non-zero only at mesh nodes where $\phi_h = \phi^{max}$. Then if $l_h \neq \mathbf{0}_{V_h}$ and $\forall \mathrm{p}_h \in \Lambda(\phi_h)$, $b(\mathrm{p}_h, l_h) = 0$, the equilibrium under gravity cannot be achieved.

Let us see if this criterion is satisfied for a few simple velocity–pressure spaces and the scenario depicted in Figure 7.3(a). We will restrict ourselves to low-order approximations (Figure 7.3(b)):

- (piecewise-constant) $\mathbb{P}_0 := \mathrm{Span}(\omega_i^0)$, with $\omega_i^0(x) := 1$ for $x_i \leq x \leq x_{i+1}$ and zero elsewhere;
- (piecewise-linear) $\mathbb{P}_1 := \mathrm{Span}(\omega_i^1)$, with $\omega_i^1(x) := \max(0, 1 - |x - x_i|)$;
- (piecewise-linear discontinuous) $\mathbb{P}_{1d} := \mathrm{Span}(\omega_{i+}^1, \omega_{i-}^1)$, with $\omega_{i+}^1(x) := \omega_i^1(x)$ for $x_i \leq x \leq x_{i+1}$ and zero elsewhere; $\omega_{i-}^1(x) := \omega_i^1(x)$ for $x_{i-1} \leq x \leq x_i$ and zero elsewhere.

T_h piecewise-constant Let us choose $\phi = \omega_0^0 + \omega_1^0 + \beta\omega_2^0$ with $0 < \beta < \frac{1}{2}$ and where ω_i^0 is the constant unitary function over the i^{th} element.

Then $\Lambda = \mathrm{Span}\left(q_0^0, q_1^0\right)$, and thus $b_h(\mathrm{p}_h, \mathbf{v}_h) = 0$ for any $\mathbf{v}_h$ vanishing on $[x_0, x_2]$. The existence of a test function $\mathbf{v}_h$ with positive integral on $[x_2, x_3]$ yet vanishing for $x \leq x_2$ would prevent the existence of an equilibrium solution.

For $V_h = \mathbb{P}_1$, we have $l_h(\omega_3^1) > 0$, and $b_h(\mathrm{p}_h, \omega_3^1) = 0$. The pair $\mathbb{P}_1 - \mathbb{P}_0$ is inconsistent. Similarly for $V_h = \mathbb{P}_{1d}$, we have $l_h(\omega_{3-}^1) > 0$, while $b_h(\mathrm{p}_h, \omega_{3-}^1) = 0$. This can be generalized to higher orders.

However, the compatibility criterion does not prohibit the $\mathbb{P}_0$–$\mathbb{P}_0$ pair.

T_h piecewise-linear We now choose $\phi = \omega_0^1 + \omega_1^1 + \beta\omega_2^1$ with $\frac{1}{2} < \beta < 1$. This means $\Lambda = \mathrm{Span}(\omega_0^1, \omega_1^1)$, and thus again, $b_h(\mathrm{p}_h, \mathbf{v}_h) = 0$ for any $\mathbf{v}_h$ vanishing on $[x_0, x_2]$.

For $V_h = \mathbb{P}_0$, $l_h(\omega_2^0) > 0$ and $b_h(\mathrm{p}_h, \omega_2^0) = 0$. Again for $V_h = \mathbb{P}_1$, $l_h(\omega_3^1) > 0$ and $b_h(\mathrm{p}_h, \omega_3^1) = 0$. For $V_h = \mathbb{P}_{1d}$, $l_h(\omega_{3-}^1) > 0$ and $b(\mathrm{p}_h, \omega_{3-}^1) = 0$. Idem for higher orders.

T_h piecewise-linear discontinuous Let $\phi = \omega_{0+}^1 + \omega_{1-}^1 + \omega_{1+}^1 + \omega_{2-}^1 + \beta\omega_{2+}^1$ with $0 < \beta < 1$. This means $\Lambda = \mathrm{Span}\left(\omega_{0+}^1, \omega_{1-}^1, \omega_{1+}^1, \omega_{2-}^1\right)$, and once again, $b_h(\mathrm{p}_h, \mathbf{v}_h) = 0$ for any $\mathbf{v}_h$ vanishing on $[x_0, x_2]$.

For $V_h = \mathbb{P}_1$ or $V_h = \mathbb{P}_{1d}$, $l_h(\mathbf{v}_h) > 0$ and $b_h(\mathrm{p}_h, \mathbf{v}_h) = 0$ for $\mathbf{v}_h := \omega_3^2$ or $\mathbf{v}_h := \omega_{3-}^2$, respectively. However the compatibility criterion does not prohibit $\mathbb{P}_0$ velocities.

Wrapping up This incompatiblity seems to arise from the fact that we are using the same discretization space for both the pressure and volume fraction fields; using a higher-order volume fraction would yield a sharper interface, and would allow this interface to stay within the range of the gradient operator stencil. However, in this case the maximal volume fraction constraint would not be able to be satisfied for every degree of freedom of the volume fraction field. In Section 7.3, we will propose to use particles to discretize the volume fraction field, and thus obtain unconditionally sharp interfaces. Alternatively, we can use one of the few discretization spaces that were found to not violate the compatibility criterion, such as the $\mathbb{P}_0$–$\mathbb{P}_0$ pair, to which the next subsection is dedicated.

7.2.2 Piecewise-constant discretization

In a manner similar to that of finite-volume methods, we consider only the average value (i.e., the value at the barycenter) of each mesh cell. Discontinuous velocity spaces are theoretically tricky to handle, are they are not subspaces of $H^1(\Omega)^d$; we refer the reader to (Pietro and Ern 2011).

Discontinuous Green formula We assume that the discrete mesh Ω_h can be decomposed as a set of disjoint polyhedras (K_i), and denote by $\mathbb{P}_{pd}(\Omega_h)$ the space of piecewise-polynomials that are of order less than p on each element K_i. Let $\mathscr{F}_\mathrm{i}$ denote the set of all interior faces of Ω_h, which we augment with an arbitrary orientation: for any $F \in F_i$, we number as K_1^F and K_2^F the adjacent cells, and define the normal $\boldsymbol{n}^F$ as pointing towards K_2^F, i.e., $\boldsymbol{n}^F = \boldsymbol{n}^{K_1^F} = -\boldsymbol{n}^{K_2^F}$.

For any field $\mathbf{v} \in \mathbb{P}_{pd}(\Omega_h)$, for any internal face F and for $\boldsymbol{x} \in F$ we denote by $\mathbf{v}_i(\boldsymbol{x})$, $i = 1$ or 2 the value at $\boldsymbol{x}$ of the restriction $\mathbf{v}$ to K_i^F, $\mathbf{v}_{|K_i}(\boldsymbol{x})$. We can then define the *jump* and *average operators* on $\mathscr{F}_\mathrm{i}$ as

$$[\![\mathbf{v}]\!] := \mathbf{v}_1 - \mathbf{v}_2 \qquad\qquad (\!|\mathbf{v}|\!) := \frac{1}{2}(\mathbf{v}_1 + \mathbf{v}_2).$$

These operators allow us to define a Stokes-like theorem for discontinuous fields. Let $\mathbf{v} \in V_0$, then

$$\begin{aligned}\int_{\mathrm{Bd}\,\Omega_h} \left\langle \mathbf{v}, \boldsymbol{n}^\Omega \right\rangle &= \sum_{K\in\Omega_h} \int_{\mathrm{Bd}\,K} \left\langle \mathbf{v}, \boldsymbol{n}^K \right\rangle - \sum_{F\in\mathscr{F}_\mathrm{i}} \int_F \left\langle \mathbf{v}_1, \boldsymbol{n}^{K_1^F} \right\rangle + \left\langle \mathbf{v}_2, \boldsymbol{n}^{K_2^F} \right\rangle \\ &= \sum_{K\in\Omega_h} \int_K \nabla\cdot\mathbf{v} - \sum_{F\in\mathscr{F}_\mathrm{i}} \int_F \left\langle [\![\mathbf{v}]\!], \boldsymbol{n}^F \right\rangle.\end{aligned}$$

Let $V_p = \mathbb{P}_p(\Omega_h)^d$, and $T_p = \mathbb{P}_p(\Omega_h)^{\mathfrak{s}_d}$. Applying our discontinuous Stokes formula to the product $\boldsymbol{\tau}\mathbf{v}$, with $\boldsymbol{\tau} \in T_p$ and $\mathbf{v} \in V_h$, and using the identity $[\![\boldsymbol{\tau}\mathbf{v}]\!] = [\![\boldsymbol{\tau}]\!] (\!|\mathbf{v}|\!) + (\!|\boldsymbol{\tau}|\!) [\![\mathbf{v}]\!]$, we get

$$\int_{\mathrm{Bd}\,\Omega_h} \left\langle \mathbf{v}, \boldsymbol{\tau}\boldsymbol{n}^\Omega \right\rangle + \sum_{F\in\mathscr{F}_\mathrm{i}} \int_F \left\langle [\![\mathbf{v}]\!], (\!|\boldsymbol{\tau}|\!)\boldsymbol{n}^F \right\rangle - \sum_{K\in\Omega_h} \int_K \mathrm{D}(\mathbf{v}) : \boldsymbol{\tau} = \sum_{K\in\Omega_h} \int_K \langle \nabla\cdot\boldsymbol{\tau}, \mathbf{v}\rangle - \sum_{F\in\mathscr{F}_\mathrm{i}} \int_F \left\langle (\!|\mathbf{v}|\!), [\![\boldsymbol{\tau}]\!]\boldsymbol{n}^F \right\rangle.$$

For the sake of brevity, we can extend the jump and average operators on $\mathscr{F}_\mathrm{a} := \mathscr{F}_\mathrm{i} \cup \mathrm{Bd}\,\Omega_h$ as $[\![\mathbf{v}]\!]_{|\mathrm{Bd}\,\Omega_h} = \mathbf{v}$ and $(\!|\mathbf{v}|\!)_{|\mathrm{Bd}\,\Omega_h} = \mathbf{v}$, and rewrite this last equation (the discontinuous Green formula) as

$$\sum_{F\in\mathscr{F}_\mathrm{a}} \int_F \left\langle [\![\mathbf{v}]\!], (\!|\boldsymbol{\tau}|\!)\boldsymbol{n}^F \right\rangle - \sum_{K\in\Omega_h} \int_K \mathrm{D}(\mathbf{v}) : \boldsymbol{\tau} = \sum_{K\in\Omega_h} \int_K \langle \nabla\cdot\boldsymbol{\tau}, \mathbf{v}\rangle - \sum_{F\in\mathscr{F}_\mathrm{i}} \int_F \left\langle (\!|\mathbf{v}|\!), [\![\boldsymbol{\tau}]\!]\boldsymbol{n}^F \right\rangle. \tag{7.12}$$

The left-hand-side of (7.12) can be interpreted as a weak expression of $\mathrm{D}(\mathbf{u})$ (with homogeneous Dirichlet boundary conditions), and the right-hand-side as a weak expression of $\nabla\cdot\boldsymbol{\tau}$.

Discrete variational formulation The bilinear forms m and s from the variational formulation given in Proposition 7.1 do not involve any spatial derivative, and can thus be left unmodified. The discontinuous Green formula (7.12) yields a discretization of the bilinear form b which can be easily verified to be consistent when the velocity solution is continuous, i.e., when $[\![\mathbf{u}]\!] = \mathbf{0}$ on $\mathscr{F}_\mathrm{i}$,

$$b_h(\mathbf{u}, \boldsymbol{\tau}) = -\sum_{F\in\mathscr{F}_\mathrm{a}} \int_F \left\langle [\![\mathbf{v}]\!], (\!|\phi\boldsymbol{\tau}|\!)\boldsymbol{n}^F \right\rangle + \sum_{K\in\Omega_h} \int_K \phi\, \mathrm{D}(\mathbf{v}) : \boldsymbol{\tau}. \tag{7.13}$$

Obviously, the rightmost term in (7.13) vanishes when considering piecewise constant polynomials, and thus b_h reduces to a sum of interfacial transfer terms.

The discretization of viscosity term, which involves a second-order derivative of the velocity, is more complex. The order of the polynomials in $\mathbb{P}_0$ is too low to allow the application of standard discontinuous Galerkin methods, such as the Symmetric Interior Penalty method (Pietro and Ern 2011, Section 4.2). Instead, we discretize our viscosity form a from Proposition 7.1

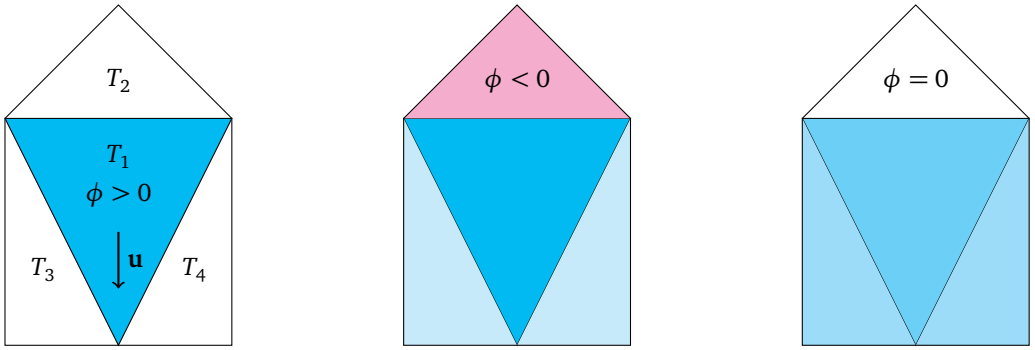


Figure 7.4: *The centered transport scheme (middle) will yield a negative volume fraction in T_2 when ϕ (cyan) and $\mathbf{u}$ are initially non-zero only on T_1. In contrast, upwind transport (right) guarantees a positive volume fraction.*

as the weak derivative of the weak derivative of $\mathbf{u}$. That is, we once again take profit of our discontinuous Green formula (7.12), and replace $a(\mathbf{u},\mathbf{v})$ with $c(\dot{\boldsymbol{\varepsilon}},\mathbf{v})$, where

$$c(\boldsymbol{\tau},\mathbf{v}) := -\int_{F\in\mathscr{F}_a} \left\langle [\![\mathbf{v}]\!], (\!|\boldsymbol{\tau}|\!)\boldsymbol{n}^F \right\rangle$$

and $\dot{\boldsymbol{\varepsilon}}$ is defined as the solution to $s(\dot{\boldsymbol{\varepsilon}},\boldsymbol{\tau}) = c(\boldsymbol{\tau},\mathbf{u})\ \forall \boldsymbol{\tau} \in T_h$. As the matrix associated to s is diagonal, $\dot{\boldsymbol{\varepsilon}}$ is trivial to compute.

Transport of the volume fraction field We now consider the second step of our semi-implicit algorithm; given a velocity field $\mathbf{u}$, computing the volume fraction field at the end of the timestep.

We can list a few desirable properties for this step:

1. admissibility: ϕ^{k+1} should take values in $[0,1]$;
2. global mass conservation: the integral of the volume fraction over the domain should remain constant, i.e., $\frac{\partial}{\partial t}\int_\Omega \phi = 0$;
3. energy conservation: in the continuous case, there is a correspondence between the work of gravity in the conservation of momentum equation and the loss of potential energy due to the motion of the volume fraction field. We would like to get a similar property in the discrete case, or at least, prevent the artificial introduction of energy.

Unfortunately, it is hard for all of those properties to simultaneously hold in the discrete setting. In our case, we prioritize the first two; in particular, if the volume fraction were to go below zero, the bilinear form m would no longer be positive and our system would be ill-posed. A correction step could be devised, at the expense of the desirable properties 2 and 3.

For stability reasons, we choose to use an implicit scheme. At the first-order, we want to solve

$$\frac{\phi^{k+1}}{\Delta_t} + \nabla\cdot\left[\phi^{k+1}\mathbf{u}\right] = \frac{\phi}{\Delta_t}. \tag{7.14}$$

Equation (7.14) can be put under weak form as $e(\phi,\psi) = f(\psi)\ \forall\psi \in L_2(\Omega)$, where, in the continuous setting,

$$e(\phi,\psi) = \frac{1}{\Delta_t}\int_\Omega \phi\psi + \int_\Omega \nabla\cdot[\phi\mathbf{u}]\psi = \frac{1}{\Delta_t}\int_\Omega \psi\phi - \int_\Omega \phi\mathbf{u}\nabla\psi$$
$$f(\psi) = \frac{1}{\Delta_t}\int_\Omega \phi^k\psi.$$

We can check that in the continuous case, the variation of potential energy $\mathscr{E}^p$ corresponds to the work of gravity (Desirable property number 3). Indeed, we get

$$\begin{aligned}\mathscr{E}^p(\phi)-\mathscr{E}^p(\phi^k) &= -\rho\int_\Omega(\phi-\phi^k)\langle \boldsymbol{x},\boldsymbol{g}\rangle = \rho\Delta_t\int_\Omega \nabla\cdot[\phi\mathbf{u}]\langle \boldsymbol{x},\boldsymbol{g}\rangle \\ &= -\rho\Delta_t\int_\Omega \phi\langle \mathbf{u},\boldsymbol{g}\rangle.\end{aligned}$$

Now, we could discretize the bilinear form e as we did for the momentum conservation variational formulation, leading to

$$\begin{aligned}e_h(\phi,\psi) &= \frac{1}{\Delta_t}\int_\Omega \phi\psi - \int_{F\in\mathscr{F}_\mathrm{a}}\left\langle [\![\phi\mathbf{u}]\!], (\!|\psi|\!)\boldsymbol{n}^F\right\rangle \\ &= \frac{1}{\Delta_t}\int_\Omega \phi\psi + \int_{F\in\mathscr{F}_\mathrm{i}}\left\langle (\!|\phi\mathbf{u}|\!), [\![\psi]\!]\,\boldsymbol{n}^F\right\rangle.\end{aligned}$$

With this discretization, we can see that the total mass is conserved (Criterion 2); indeed

$$\frac{1}{\Delta_t}\int_{\Omega_h}\phi-\phi^k = e_h(\phi,\mathbb{1}) - \int_{F\in\mathscr{F}_\mathrm{i}}\left\langle (\!|\phi\mathbf{u}|\!), \underbrace{[\![\mathbb{1}]\!]}_{0}\right\rangle - f_h(\mathbb{1}) = 0.$$

However, looking at the difference in potential energy, we get, denoting by $\mathbf{x}_K\in V_0$ the field associating to each mesh cell the position of its barycenter

$$\begin{aligned}\frac{\mathscr{E}^p(\phi)-\mathscr{E}^p(\phi^k)}{\rho\,\Delta_t} &= -\int_\Omega(\phi-\phi^k)\langle \boldsymbol{x},\boldsymbol{g}\rangle = -\int_\Omega(\phi-\phi^k)\langle \mathbf{x}_K,\boldsymbol{g}\rangle \\ &= \underbrace{f_h(\langle \mathbf{x}_K,\boldsymbol{g}\rangle) - e_h(\phi,\langle \mathbf{x}_K,\boldsymbol{g}\rangle)}_{0} - \int_{F\in\mathscr{F}_\mathrm{a}}\left\langle [\![\phi\mathbf{u}]\!], \left\langle (\!|\mathbf{x}_K|\!),\boldsymbol{g}\right\rangle \boldsymbol{n}^F\right\rangle.\end{aligned}$$

On the other hand,

$$\begin{aligned}\int_\Omega \phi\langle \mathbf{u},\boldsymbol{g}\rangle &= \sum_K\int_K\langle \phi\mathbf{u}, \nabla\cdot[\langle \boldsymbol{x},\boldsymbol{g}\rangle\,\mathbb{I}]\rangle = \sum_K\int_{\mathrm{Bd}\,K}\left\langle \phi\mathbf{u},\boldsymbol{n}^K\right\rangle\langle \boldsymbol{x},\boldsymbol{g}\rangle \\ &= \sum_{F\in\mathscr{F}_\mathrm{a}}\left\langle [\![\phi\mathbf{u}]\!],\boldsymbol{n}^F\right\rangle\langle \boldsymbol{x},\boldsymbol{g}\rangle.\end{aligned}$$

The difference between these two terms is $\sum_{F\in\mathscr{F}_\mathrm{a}}\left\langle [\![\phi\mathbf{u}]\!],\boldsymbol{n}^F\right\rangle\left\langle \boldsymbol{x}-(\!|\mathbf{x}^K|\!),\boldsymbol{g}\right\rangle$, and is non-zero in the general case. Criterion 3 is thus not satisfied. This discrepancy can be understood as the transported material "jumping" from one barycenter to one other, regardless of their respective proximity to the face.

Moreover, this is a centered scheme; as visible in the expression of e_h, the flux between cells is defined by the average $(\!|\phi\mathbf{u}|\!)$. If one cell is initially empty, one of its neighbor is not, and the velocity points toward the non-empty cell, then the initially empty cell may end up with a negative value of ϕ. For instance, consider the T_2 cell in Figure 7.4, where $\phi\mathbf{u}=\boldsymbol{e}_y$ on T_1 and $\phi=0$ elsewhere. Criterion 1 is thus not satisfied either.

To remediate to this problem, we use an upwind scheme. The flux between cells will then be computed considering only the volume fraction of the upwind cell. As we are using an implicit scheme, if this value becomes zero the flux will vanish, and thus the volume fraction will not become negative. For any internal face F, let ϕ^{upw} denote the value of ϕ the value on the upstream cell, that is, ϕ_i such that $\left\langle \boldsymbol{n}^{K_i},(\!|\mathbf{u}|\!)\right\rangle\geq 0$. We want the flux between cells to be $\phi^{\mathrm{upw}}(\!|\mathbf{u}|\!)$;

the upwind bilinear form e^{upw} can thus be obtained as

$$\begin{aligned} e_h^{\text{upw}}(\phi,\psi) &= e_h(\phi,\psi) + \sum_{\mathscr{F}_i} [\![\phi]\!] \left\langle \phi^{\text{upw}} (\!|\mathbf{u}|\!) - (\!|\phi\mathbf{u}|\!), \boldsymbol{n}^F \right\rangle [\![\psi]\!] \\ &= e_h(\phi,\psi) + \frac{1}{2} \sum_{\mathscr{F}_i} [\![\phi]\!] \left(\left| \left\langle (\!|\mathbf{u}|\!), \boldsymbol{n}^F \right\rangle \right| - \frac{1}{2} \left\langle [\![\mathbf{u}]\!], \boldsymbol{n}^f \right\rangle \right) [\![\psi]\!] \\ &= \frac{1}{\Delta_t} \int_\Omega \phi\psi + \int_{F\in\mathscr{F}_i} [\![\psi]\!] \left(\left\langle (\!|\phi|\!)(\!|\mathbf{u}|\!), \boldsymbol{n}^F \right\rangle + \frac{1}{2} [\![\phi]\!] \left| \left\langle (\!|\mathbf{u}|\!), \boldsymbol{n}^F \right\rangle \right| \right). \end{aligned}$$

With a similar reasoning to that of the centered case, we can verify that the upwind discretization verify Criterion 2 on the total mass conservation. However, energy conservation is not better than for the centered discretization. Criterion 1 and 2 being our priorities, we still choose to use the upwind bilinear form and take ϕ^{k+1} as the solution of $e_h^{\text{upw}}(\phi,\psi) = l(\psi)\ \forall\psi \in \mathbb{P}_0(\Omega_h)$.

Inertial terms Finally, it remains to deal with the velocity advection term $(\mathbf{u}\cdot\nabla)\mathbf{u}$ in the conservation of momentum equation. As the velocities are discontinuous, the characteristics method cannot be used (the morphism X is no longer locally invertible). Pietro and Ern (2011, Chapter 6) demonstrate that discretizing implicitly the transport trilinear form $t(\mathbf{w},\mathbf{u},\mathbf{v})$,

$$t(\mathbf{w},\mathbf{u},\mathbf{v}) := -\int_\Omega \langle (\mathbf{w}\cdot\nabla)\mathbf{u}, \mathbf{v}\rangle,$$

using to the so-called *Temam* modification preserves the kinetic energy of the system. However, this modification is only valid for incompressible flows, and yields an asymmetric system; it is therefere not applicable in our case. In the end, we simply used an explicit version of the upwind transport scheme described in the previous paragraph, which we found yielded in practice a good compromise between stability and conservation of kinetic energy. Thus, we define $u(\mathbf{u}^k)$ in the computation of the l linear form from Property 7.1 such that

$$\begin{aligned} \int_{\Omega_h} \left\langle u(\mathbf{u}^k), \boldsymbol{v} \right\rangle &:= \int_{\Omega_h} \left\langle \mathbf{u}^k, \boldsymbol{v} \right\rangle - t_h(\mathbf{u}^k, \mathbf{u}^k, \mathbf{v}) \\ t_h(\mathbf{w},\mathbf{u},\mathbf{v}) &:= -\sum_{F\in\mathscr{F}_i} \left\langle [\![\mathbf{u}]\!], \left\langle (\!|\mathbf{w}|\!), \boldsymbol{n}^F \right\rangle (\!|\mathbf{v}|\!) + \frac{1}{2} \left| \left\langle (\!|\mathbf{w}|\!), \boldsymbol{n}^F \right\rangle \right| [\![\mathbf{v}]\!] \right\rangle. \end{aligned}$$

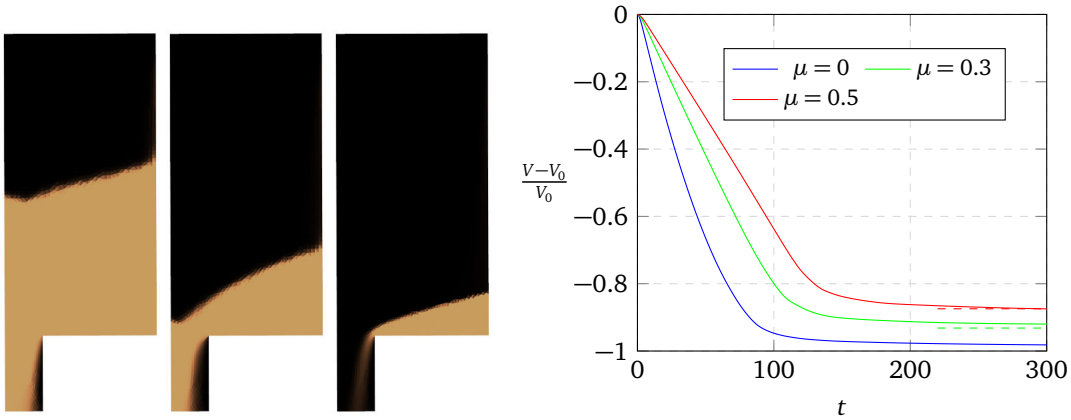
Final discrete system The final form of the variational formulation from Property 7.1 after $\mathbb{P}_0$–$\mathbb{P}_0$ discretization is thus:
Find $\underline{\boldsymbol{u}} \in \mathbb{R}^{\mathfrak{v}}$, $\underline{\boldsymbol{\lambda}}, \underline{\boldsymbol{\gamma}} \in \mathbb{R}^{n\,\mathfrak{s}_d}$,

$$\begin{cases} (M + C^\intercal S^{-1} C)\underline{\boldsymbol{u}} = B^\intercal \underline{\boldsymbol{\lambda}} + \underline{\boldsymbol{l}} \\ S\underline{\boldsymbol{\gamma}} = B\underline{\boldsymbol{u}} + \underline{\boldsymbol{k}} \\ (\underline{\boldsymbol{\gamma}}, \underline{\boldsymbol{\lambda}}) \in \mathscr{D}\mathscr{P}(\mu), \end{cases} \tag{7.15}$$

where M and S are diagonal, positive matrices. As all the integrals of the variational formulation vanish on empty mesh cells, we can only consider those in which the volume fraction is non-zero, and thus M is positive-definite. This has also the advantage of drastically reducing the size of the numerical system in some scenarios. Alternatively, we can take into account the presence of a light phase in the computation of the density in each cell, as in (Lagrée et al. 2011). In both cases, System (7.15) can be then solved in the exact same manner as for our dense flows; see Section 6.4.

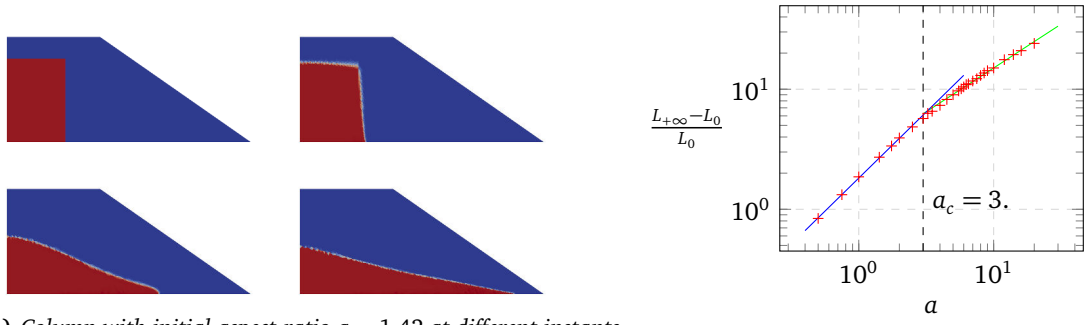
7.2.3 Results

We attempted to validate our piecewise-constant discretization strategy by looking at some empirical laws reported in the literature. Just like in the dense case, the finite-element discretization was done with the Rheolef library (Saramito 2015).



(a) *Volume fraction field at chosen instants* **(b)** *Volume loss over time*

Figure 7.5: *Evolution of the volume fraction of grains in a discharging silo.*



(a) *Column with initial aspect-ratio $a = 1.42$ at different instants.* **(b)** *Run-out length versus aspect-ratio a*

Figure 7.6: *Collapse of a 2D granular column using the $\mu(I)$ rheology with for $\mu_S = 0.32$, $\mu_D = 0.6$, $I_0 = 0.4$ and $Re = 10^4$.*

Silo discharge The first experiment is quite similar to the one that we already studied for dense flows in Section 6.5.3. However, we are no longer interested in the steady discharge regime, but rather in the evolution of the volume of remaining matter in the 2D silo through time, as illustrated in 7.5(a). Staron et al. (2012) recall that one distinguishing feature of an hourglass is that the sand flow rate remains constant for most of the discharge process. This contrasts with clepshydras, which contain a standard Newtonian liquid and whose discharge rate decreases with time. Figure 7.5(b) demonstrates that we retrieve this behavior; for $\mu = 0$, the discharge rate $\frac{\mathrm{d}V}{\mathrm{d}t}$ is mostly decreasing, while for $\mu = 0.3$ and $\mu = 0.5$, it remains constant as long as the volume V stays in $[0.9V_0, 0.2V_0]$, where V_0 denotes the initial volume of grains in the silo. The dashed horizontal lines in Figure 7.5(b) represent the theoretical final volume of grains computed from the Mohr–Coulomb rest angle corresponding to each friction coefficient.

Granular column collapse Another classical experiment that we can now run thanks to our volume fraction field concerns the 2D collapse of a granular column, as depicted in Figure 7.6(a). Note that this scenario is known to be approachable by continuum models; Ionescu et al. (2015) managed to quantitatively match experiments using the $\mu(I)$ rheology. An empirical law relates the run-out length of the flow L_∞ (that is, the maximum abscissa reached by the grains) to the initial aspect-ratio of the column (that is, the ratio a of its height H_0 to its width L_0). The

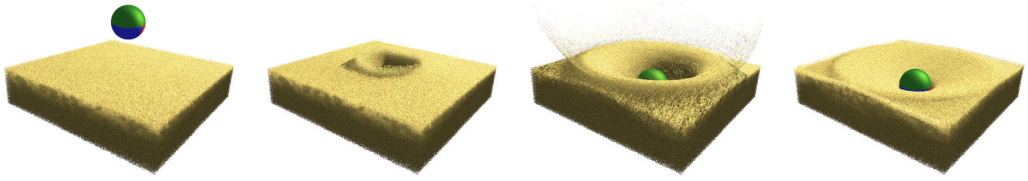


Figure 7.7: *A steel ball impacts a granular bed. Unlike previous approaches, our fully resolved (non-linearized) Drucker-Prager rheology allows us to retrieve a perfectly round crater.*

experiments of Lajeunesse et al. (2005) led to the distinction of two regimes:

$$\frac{L_\infty - L_0}{L_0}(a) \propto \begin{cases} a & \text{if } a < a_c \\ a^{\frac{2}{3}} & \text{if } a > a_c, \end{cases}$$

with $a_c \sim 3$. The numerical simulations of Staron and Hinch (2005), using DEM, and Lagrée et al. (2011), using the $\mu(I)$ rheology, yielded similar scaling laws, although with different critical aspect ratios, $a_c \sim 2$ and 7. Several other authors have performed similar experiments both experimentally and numerically, see (Dunatunga and Kamrin 2015, Table 3) for a summary of their results. Using our method with the physical parameters from Lagrée et al. (2011), we obtained the scaling law plotted in Figure 7.6(b):

$$\frac{L_\infty - L_0}{L_0}(a) \sim \begin{cases} 1.82a^{1.1} & \text{if } a < 3 \\ 2.8a^{0.73} & \text{if } a > 3. \end{cases}$$

The exponents that we obtained in both regimes seem to be a bit on the high side, though they remain within the range of reported values in the numerical simulation literature, especially when taking into account the results of (Mast et al. 2014). Note that at the characteristic sizes chosen for our spatial and temporal discretization, the choice of the advection scheme for the volume fraction and velocity fields retains a non-negligible influence on the final scaling law.

Discussion This fully mesh-based discretization strategy suffers from several drawbacks. We observe diffusion of the volume fraction field over time, as a lot of cells retain slightly-above-zero volume fraction even once the bulk of the material has moved away. The explicit transport scheme for the inertial term also prohibits taking large timesteps, making the simulation process quite slow. In the next section we propose to use a different strategy and resort to an hybrid method, which will discretize the volume fraction field using a set of particles. These particles will also be used to transport other quantities, such as the inertial terms.

7.3 Material Point Method

The basic idea behind MPM is to consider that the whole mass of the material is condensed at a finite number of material points. The volume fraction field ϕ can therefore be expressed as $\phi(\boldsymbol{x}, t) = \sum_p V_p \delta(\boldsymbol{x} - \boldsymbol{x}_p(t))$, where δ is the Dirac distribution, and V_p, $\boldsymbol{x}_p$ are the volume and position of the p^{th} particle, respectively. The total volume of granular reads $V_{\text{tot}} = \int_\Omega \phi(t) = \sum_p V_p$.

This representation is particularly adapted to the FEM formalism, where quantities are evaluated in weak-form, by multiplying them by a test function and integrating over a domain. For instance, for $\mathbf{v} \in L_2(\Omega)$, we have $\int_\Omega \phi(t)\mathbf{v} = \sum_p V_p \mathbf{v}(\boldsymbol{x}_p(t))$. In this sense, the Material Point Method can be seen as a quadrature rule for which the quadrature points are given by the particles positions, and the corresponding weights by the particles volumes.

Related work The MPM method was introduced by Sulsky (1994) for the simulation of history-dependent elastic materials. Storing information on particles made tracking the position of each material point through time trivial, avoiding solving complex transport equations.

MPM was quickly recognized as relevant for the simulation of granular materials. Wieckowski, Youn, et al. (1999) proposed a first explicit algorithm for the integration of an elasto-plastic material subject to the non-associated Drucker–Prager flow rule [1] in 2D, and recently extended it to the 3D setting (Wieckowski and Pawlak 2015). S. Bardenhagen et al. (2000) proposed to use MPM for simulating the dynamics of many 2D deformable grains with sliding and rolling contacts. Konagai and Johansson (2001) coupled the Mohr–Coulomb yield criterion with a nodal finite-difference scheme to simulate 2D granular flows. More recently, Mast et al. (2014) performed 2D granular column collapse simulations using MPM with the Matzuo–Nakai yield surface. Dunatunga and Kamrin (2015) proposed a MPM method with two distinct regimes depending on whether the volume fraction is above or below a critical value, again for 2D flows.

Stomakhin, Schroeder, Chai, et al. (2013) popularized MPM to the Computer Graphics community, coupling this method with an elasto-plastic model to produce visually-striking snow simulations. This gave rise to a variety of related methods modeling a wide range of physical phenomenons, such as visco-elasto-plastic foams (Ram et al. 2015; Yue et al. 2015), phase changes (Stomakhin, Schroeder, Jiang, et al. 2014), granular materials (e.g., concurrently to the work presented below, Klar et al. 2016), and more (Jiang, Schroeder, Teran, et al. 2016). Klar et al. (2016) follow the approach of Wieckowski, Youn, et al. (1999) with one notable difference: while both prevent plastic compression, Klar et al. (2016) allow plastic expansion, performing Alart–Curnier-like projections onto the Drucker–Prager set of admissible stresses.

Most MPM methods are explicit (Dunatunga and Kamrin 2015; Wieckowski, Youn, et al. 1999; Yue et al. 2015), or semi-implicit with respect to the elastic energy, but not w.r.t. the plasticity (Ram et al. 2015; Stomakhin, Schroeder, Chai, et al. 2013). Klar et al. (2016) and Mast et al. (2014) propose to use a "return-mapping" algorithm to perform semi-implicit integration taking into account the plasticity. In practice, this means a Newton algorithm prone to falling into local minima. In contrast, our semi-implicit integration procedure will leverage our DCFP solvers that showed good empirical convergence.

Extensions of MPM using different discretization strategies for the volume fraction field have also been proposed; for instance, the Generalized Interpolation Material Point Method (GIMP; S. G. Bardenhagen and Kober 2004) use arbitrary indicator functions for the particles.

7.3.1 Application to our variational formulation

Taking advantage of the expression of ϕ as a sum of Dirac distributions, we can now rewrite the bilinear and linear forms from the variational formulation of Property 7.1 as:

$$
\begin{aligned}
m(\mathbf{u},\mathbf{v}) &:= \sum_p \frac{\rho}{\Delta_t} V_p \left\langle \mathbf{u}(\boldsymbol{x}_p^k), \mathbf{v}(\boldsymbol{x}_p^k) \right\rangle & s(\boldsymbol{\gamma},\boldsymbol{\tau}) &:= \int_\Omega \boldsymbol{\gamma} : \boldsymbol{\tau} \\
a(\mathbf{u},\mathbf{v}) &:= \sum_p V_p \eta \, \mathrm{D}(\mathbf{u})(\boldsymbol{x}_p^k) : \mathrm{D}(\mathbf{v})(\boldsymbol{x}_p^k) & b(\boldsymbol{\tau},\mathbf{u}) &:= \sum_p V_p \boldsymbol{\tau}(\boldsymbol{x}_p^k) : \mathrm{D}(\mathbf{u})(\boldsymbol{x}_p^k) \\
l(\mathbf{v}) &:= \sum_p \rho V_p \left\langle \left(\frac{u(\mathbf{u}^k)}{\Delta_t} + \boldsymbol{g} \right), \mathbf{v}(\boldsymbol{x}_p^k) \right\rangle & k(\boldsymbol{\tau}) &:= \int_\Omega \frac{\phi_{\max}}{\Delta_t} \frac{\mathbb{I}}{d} : \boldsymbol{\tau} - \sum_p \frac{V_p}{\Delta_t} \frac{\mathbb{I}}{d} : \boldsymbol{\tau}(\mathbf{x}_p)
\end{aligned}
$$

and where $\boldsymbol{x}_p^k$ denotes the position of the p^{th}th particle at timestep k.

7.3.2 Grid–particles transfers

In the context of MPM, the semi-implicit integration algorithm reads as follow:

[1] which is expressed as the association of the Drucker–Prager yield surface with a plastic incompressibility criterion, that is, $\dot{\boldsymbol{\varepsilon}}_p \in \mathcal{N}_{\mathscr{B}_{\mathbf{T}}(-\mu\sigma_{\mathrm{N}})}(\boldsymbol{\sigma}_{\mathbf{T}})$

1. solve the variational formulation (7.10), where the advected velocity field $u(\mathbf{u}^k)$ is reconstructed from the particles;
2. deduce the new particle velocities $\boldsymbol{v}_p^{k+1}$ from the velocity field $\mathbf{u}^{k+1}$, and move the particles accordingly.

Two steps remain to be defined; first, the computation of $u(\mathbf{u}^k)$ (particles-to-grid transfer), and the computation of the particles velocities (grid-to-particle transfer). We mentioned in Section 5.2.4 different strategies for doing these transfers, such as PIC and FLIP. Both of those methods compute $\mathbf{u}^{p\mapsto g} := u(\mathbf{u}^k)$ at each mesh node as a weighted interpolation of the velocities of the surrounding particles, as per Equation (5.16). As we are using the FEM formalism however, the natural way to compute $\mathbf{u}^{p\mapsto g}$ would be to solve

$$\frac{\Delta_t}{\rho} m(\mathbf{u}^{p\mapsto g}, \mathbf{v}) = \int_\Omega \phi \langle \mathbf{u}, \mathbf{v} \rangle = \sum_p V_p \left\langle \boldsymbol{v}_p^k, \mathbf{v}(\boldsymbol{x}_p^k) \right\rangle,$$

leading to a linear system $M \underline{\boldsymbol{u}}^{p\mapsto g} = \underline{\boldsymbol{v}}$, where

$$M_{i,j} := \sum_p V_p \left\langle \mathbf{V}_i(\boldsymbol{x}_p^k), \mathbf{V}_j(\boldsymbol{x}_p^k) \right\rangle$$

and $\boldsymbol{v}_i := \sum_p V_p \boldsymbol{v}_p^k \mathbf{V}_i(\boldsymbol{x}_p^k)$. M is called the *consistent* mass matrix.

In contrast, the standard PIC/FLIP update (5.16) can be understood as solving a diagonal system $\tilde{M} \underline{\boldsymbol{u}}^{p\mapsto g} = \underline{\boldsymbol{v}}$, with

$$\tilde{M}_{i,j} := \sum_p V_p \left\langle \mathbf{V}_i(\boldsymbol{x}^k + p), \mathbb{1} \right\rangle \delta_i^j \tag{7.16}$$

Assume that $\forall \boldsymbol{x} \in \Omega$, $\sum \mathbf{V}_j(\boldsymbol{x}) = \mathbb{1}_{\mathbb{R}^d}$, as is the case for standard Lagrange FEM. Then $\tilde{M}_{i,j} = \sum_j M_{i,j}$, and $\tilde{M}$ is called the *diagonalized* or *lumped* mass matrix. Alternatively, $\tilde{M}$ can be understood as the result of using a trapezoidal approximation of the integral in the bilinear form m.

Computing $\mathbf{u}^{p\mapsto g}$ using M instead of $\tilde{M}$ yields much better kinetic energy conservation, alleviating the main drawback of PIC (Burgess et al. 1992). Indeed, the kinetic energy discrepancy is $O(\Delta_t |\mathbf{u}|)$ for the consistent mass matrix, instead of $O(|\mathbf{u}|)$ for the lumped mass matrix. However, we can see from the interpretation of the particles as quadrature points that M will be indefinite if there are not enough particles in one given mesh cell. Using the consistent mass matrix is therefore rarely possible in practice.

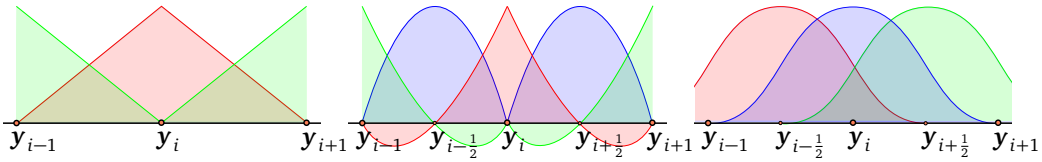
Moreover, FLIP is not convenient in our case. Indeed, consider a particular falling on a flat granular bed; we want the particle to stop as soon as it reaches the ground. As the granular bed is not moving, the velocity delta $(\mathbf{u}^{k+1} - \mathbf{u}^k)$ will be zero inside the ground; the FLIP update rule (5.19) will thus not modify the falling particle's velocity, and this particle will dive inside the ground. Using a weighted PIC/FLIP update rule does not fully correct this problem.

Instead, we will opt for the strategy recently proposed by Jiang, Schroeder, Selle, et al. (2015), the Affine Particle-in-Cell (APIC) method. The original PIC velocity update rule (5.17) is then used; in our falling particle scenario, this means that the particle velocity will become equal to the ground velocity ($\mathbf{0}$) as soon as it reaches it. The kinetic energy loss due to the lumped-mass matrix is alleviated by storing information about the velocity gradient on the particles, and using this information during the particle-to-grid transfer step. Instead of computing the velocity at mesh nodes as a weighted average of particle velocities, they are computed as the weighted of average of the node velocities *as seen by the particles*.

The exact procedure depends on the shape functions chosen for the velocity field; in the case of d-linear shape functions, with $\mathbf{V}_{d\,(i-1)+k} = \omega_i^v \boldsymbol{e}_k$, the expression boils down to

$$\underline{\boldsymbol{u}}^{p\mapsto g} = \tilde{M}^{-1} \underline{\boldsymbol{v}}^{\text{APIC}} \tag{7.17}$$

$$\underline{\boldsymbol{v}}^{\text{APIC}}_{3(i-1)+k} = \sum_p V_p (\boldsymbol{v}_{p,k}^k \omega_i^v(\boldsymbol{x}_p^k) + \boldsymbol{c}_{p,k}^k (\boldsymbol{y}_i - \boldsymbol{x}_p^k)) \tag{7.18}$$



(a) *Linear* ($\mathbb{Q}_1$) **(b)** *Quadratic* ($\mathbb{Q}_2$) **(c)** *Cubic spline*

Figure 7.8: *A few possible choices for MPM velocity shape functions*

where $\boldsymbol{y}_i$ denotes the position of grid node i and $\boldsymbol{c}_{p,k}$ the k^{th} row of $\boldsymbol{c}_p^k$, the 3×3 velocity gradient matrix computed at the end of the previous step as

$$\boldsymbol{c}_p^k = (\nabla \mathbf{u})(t^k, \boldsymbol{x}_p^k) = \sum_i \mathbf{u}(t^k, \boldsymbol{y}_i)(\nabla \omega_i^v)(\boldsymbol{x}_p^k). \tag{7.19}$$

See (Jiang, Schroeder, Selle, et al. 2015, Sections 5 and 6) for more details.

Adaptive resampling Even when using the lumped mass matrix to avoid indefinite systems, for the MPM discretization to be justified the particles needs to remain well-distributed over the mesh cells. However, if no resampling is performed, expansion of the material will inevitably lead to a scarcity of particles. Conversely, if two many particles becomes concentrated in a single cell, the system might become overconstrained when using particle-based stress shape functions.

For these reasons, we follow the split/merge heuristics of Narain, Golas, et al. (2010, Section 3). To each particle is associated an ellipsoid representing its occupied volume. Such an ellipsoid with semi-axis $(\boldsymbol{a}_{p,i})_{1\leq i\leq d}$ defines a symmetric, positive-definite tensor $F_p := \sum \boldsymbol{a}_{p,i}\boldsymbol{a}_{p,i}^\intercal$, whose change in time satisfies

$$\frac{\mathrm{d}F_p}{\mathrm{d}t} = (\mathrm{D}(\mathbf{u}) + \mathrm{W}(\mathbf{u}))(\boldsymbol{x}_p(t))F_p + F_p(\mathrm{D}(\mathbf{u}) - \mathrm{W}(\mathbf{u}))(\boldsymbol{x}_p(t)). \tag{7.20}$$

At each timestep, we assume $\boldsymbol{x}_p(t)$ constant and compute the solution to (7.20) using matrix exponentiation.

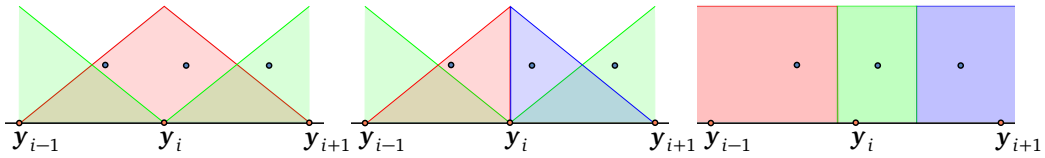
Particles are then split if one axis of the ellipsoid becomes much longer than the others. The reverse operation is performed when two close particles can be merged into a more isotropic one.

Still following Narain, Golas, et al. (2010), these frames are useful not only at simulation time, but also at the rendering stage. Indeed, they define volumes from which the passively-advected grain samples are not allowed to escape. This ensures that a rendering sample cannot drift to a place where no simulated particle is present, i.e., to a place where the velocity field has no physical meaning.

7.3.3 Shape functions

The Material Point Method prescribes how the volume fraction field should be discretized, but suitable discretizations for the velocity and stress fields remain to be chosen.

Background mesh The background mesh used for the computation of the momentum conservation equation is typically chosen to be a regular grid (which allows efficiently locating particles), but triangle or tetrahedrical-based meshes have also been used (e.g., Wieckowski and Pawlak 2015; Wieckowski, Youn, et al. 1999). Stomakhin, Schroeder, Jiang, et al. (2014) use a staggered grid to simplify the handling of incompressibility constraints.



(a) *Linear ($\mathbb{Q}_1$)* **(b)** *Discontinuous linear ($\mathbb{Q}_{1d}$)* **(c)** *Particle-based*

Figure 7.9: *A few possible choices for MPM stress shape functions*

Velocity field Historically, linear ($\mathbb{P}_1$) or multi-linear ($\mathbb{Q}_1$) elements (Figure 7.8(a)) have been favored for the discretization of the velocity field (S. Bardenhagen et al. 2000; Dunatunga and Kamrin 2015; Mast et al. 2014; Sulsky 1994; Wieckowski and Pawlak 2015; Wieckowski, Youn, et al. 1999). However, higher-order shape functions have also been proposed: d-quadratic ($\mathbb{Q}_2$; Figure 7.8(b)) elements (S. Andersen and L. Andersen 2010), which feature only C^0 continuity, or quadratic and cubic splines (Steffen et al. 2008; Figure 7.8(c)), which boast C^1 continuity but span over several grid elements, have non-zero values at more than one node. Due to their smoothness, cubic splines have been especially popular for Computer Graphics applications, and are used for instance in (Klar et al. 2016; Ram et al. 2015; Stomakhin, Schroeder, Chai, et al. 2013; Yue et al. 2015).

Steffen et al. (2008) argue that C^1 continuity is required to avoid so-called *cell-crossing artifacts*. Indeed, without this property, the test velocity gradient field $\mathrm{D}(\mathbf{v})$ is discontinuous, and the integration error increases drastically when a particle crosses a cell-boundary during a timestep. They showed that on some simple test cases linear shape functions did not yield convergence when the spatial resolution was increased, while quadratic and cubic splines performed better. However, for rough discretizations (i.e., while still within the convergence regime of linear shape functions), using higher-order splines did not provide much more accuracy. S. Andersen and L. Andersen (2010) compared $\mathbb{Q}_1$, $\mathbb{Q}_2$ elements and cubic splines on scenarios less prone the cell-crossing instabilities, and observed once again that cubic splines were not much more accurate that linear elements, yet helped to better resolve collisions. $\mathbb{Q}_2$ (and serendipity) elements yielded much lower integration error, but are subject to a major issue impeding their use in practice: as the $\mathbb{Q}_2$ (or $\mathbb{P}_2$) shape functions can take negative values, when there are not enough particles per cell they may yield negative coefficients in the lumped mass matrix $\tilde{M}$.

As our primary application for this hybrid method is Computer Graphics, we will prioritize computational efficiency over spatial convergence properties. This will drive our choice to use d-linear shape functions, as this discretization strategy will lead to much sparser matrices (there is less overlap between the different shape functions than for, say, quadratic or cubic splines), and thus much more efficient numerical solvers.

Stress field In contrast, MPM is pretty opinionated about the way the stress field should be discretized, and a large majority of the methods from the literature choose to make the stress shape functions coincide with the particles. That is, the basis (ω^τ) is chosen so that $\omega_q^\tau(\boldsymbol{x}_p) = \delta_q^p$, which amounts to associating to each particle its own stress $\boldsymbol{\sigma}_p$, as in Figure 7.9(c). In most methods, the stress field is evaluated only at the particle positions, so there are no need to precisely define the shape of the stress basis elsewhere. However, there are a few reasons for looking for alternatives in our case.

- First, a peculiarity of our method is that we define a critical volume fraction $\phi_{\max}$. This manifests itself by the presence in the linear form k of the variational formulation of the integral $V(\boldsymbol{\tau}) := \int_\Omega \frac{\mathbb{I}}{d} : \boldsymbol{\tau}$. Computing $k(\mathbf{T}_p)$ thus requires being able to evaluate the "volume" $V(\omega_p^\tau)$ of each stress shape function. This volume has an actual physical meaning: if it is larger than V_p, the flow may still compress, while if it is equal to or below V_p, the critical density has already been reached. We must thus be able to evaluate the volume

of each (ω_p^τ), the amount of space available for each particle to fill-up. A first solution is to compute these volumes from the particles positions; for instance, by constructing a Voronoi diagram, or by computing the distance of each particle to its k-nearest-neighbors. Another solution, more in line with the MPM philosophy, and proposed by Dunatunga and Kamrin (2015), is to track these volumes over time. Actually, this quantity is readily available as the volume of the frames used for the particle resampling strategy of Narain, Golas, et al. (2010) presented is the previous section; indeed, $V(\omega_p^\tau) = \sqrt{\det F_p}$, where F_p is given by Equation (7.20).

- A second drawback concerns the number of constraints of the system. As we saw in Chapter 6 for dense flows, the number of stress shape functions can be directly linked to the number of constraints in our system. A higher number of constraints leads to a more expensive system to solve, so it might be beneficial to use less than one stress shape function per particle. Moreover, when using multilinear shape functions for the velocities, the number of degrees of freedom of the system is reduced to d per grid node, and using one constraint per particle may lead to an overconstrained system. Indeed, Mast et al. (2014) describes a kinematic locking phenomenon for nearly incompressible systems when using multilinear velocities and particle-based stresses. A similar phenomenon can manifest itself for our unilateral incompressiblity criterion; it suffices for a single particle in a cell to be maximally compacted (i.e., $V(\omega_p^\tau) = V_p$) to prevent negative divergence of the velocity inside the cell, even if the other particles of the cell have still room for compression.

We thus advocate considering multilinear shape functions for the stresses as well; in many cases, this will lead to much faster solving. Note however that this choice can also create artifacts in some scenarios.

- Stable configurations maintained by cohesion are hard to achieve. Indeed, as Remark 7.1 imposes that cohesion vanishes at nodes where $\phi < \phi_{\max}$, a structure maintained by cohesion will progressively erode unless particles are in a very special configuration.
- Another failure case for multilinear stresses in when a big lump of material falls in a neighboring cell to light particles. The momentum of the falling heavy particles will be transmitted to the light ones, resulting in a disturbing kicking effect (Figure 7.10). This scenario is prone to appear in the bottom compartment of an hourglass, for instance.

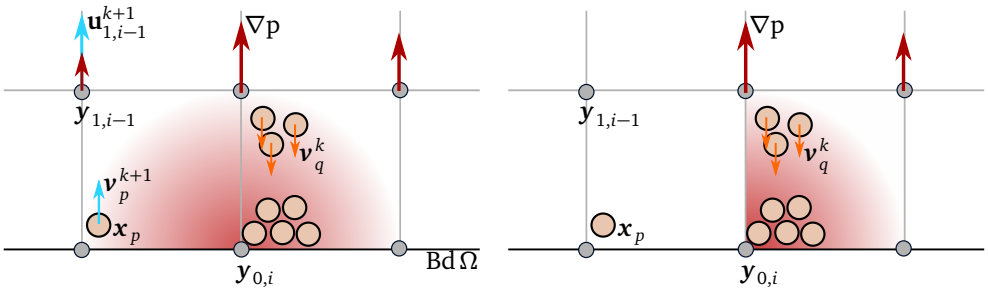
In both cases, reverting to particle-based stresses will alleviate the problem. An alternative solution is to consider multilinear-discontinuous $\mathbb{Q}_{1d}$ stress shape functions, as in Figure 7.9(b), at the cost of having to solve 2^d constraints per (active) grid cell. As this space does not depend on particle history and maintains a stable constraints–degrees-of-freedom ratio, $\mathbb{Q}_{1d}$ is less prone to kinematic-locking than particle-based stresses, but may still induce grid artifacts in the presence of cohesion. Figure 7.11 illustrates that using discontinuous multilinear shape functions allows simulating the bottom compartment of an hourglass.

Compability criterion Finally, the choice of discrete basis for the velocity and stress still haves to satisfy a compability criterion to allow the expression of equilibria, as argued is Section 7.2.1. We found $\mathbb{Q}_1$–$\mathbb{Q}_1$, $\mathbb{Q}_1$–$\mathbb{Q}_{1d}$ and $\mathbb{Q}_1$–*particle* to behave well, however $\mathbb{P}_1$–$\mathbb{P}_{1d}$ on a regular tetrahedral grid failed to reach an equilibrium state after a column collapse. In this case, switching to piecewise discontinuous stresses, $\mathbb{P}_1$–$\mathbb{P}_{1d}$, was sufficient to restore stability, as the cost of a much increased number of constraints.

7.3.4 Numerical resolution

Once our velocity and stress basis functions have been chosen, the procedure described in Section 6.3 can be applied so that we end up with the discrete system (7.21),

$$\begin{cases} (\tilde{M} + A)\underline{\boldsymbol{u}} = B^\intercal \underline{\boldsymbol{\lambda}} + \underline{\boldsymbol{l}} \\ \underline{\boldsymbol{\gamma}} = B\underline{\boldsymbol{u}} + \underline{\boldsymbol{k}} \\ (\underline{\boldsymbol{\gamma}}_{[i]}, \underline{\boldsymbol{\lambda}}_{[i]}) \in \mathscr{D}\mathscr{P}(\mu) \quad \forall 1 \leq i \leq n. \end{cases} \tag{7.21}$$



(a) *Linear* ($\mathbb{Q}_1$) **(b)** *Linear discontinuous* ($\mathbb{Q}_{1d}$)

Figure 7.10: *A clump of particles falls on the ground near an isolated one.* ***(a)*** *When using multilinear stresses, the pressure (red) at the node* $\boldsymbol{y}_{0,i}$ *induces an upwards force and velocity at the node* $\boldsymbol{y}_{1,i-1}$*, kicking the isolated particle* $\boldsymbol{x}_p$ *away.* ***(b)*** *When using discontinuous multilinear stresses, this force vanishes.*

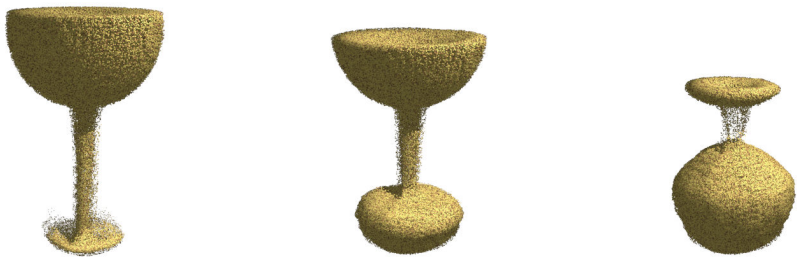


Figure 7.11: *Hourglass simulation using discontinuous stress shape functions*

The solvers presented in Section 6.4 can then be used; however, they were found to be quite slow in 2D, and thus unlikely to scale up to three-dimensional problems. Moreover, as we have taken the shear yield stress of the Drucker–Prager law to be zero, System (7.21) is now exactly a DCFP, though one were the inverse of the stiffness matrix, $(\tilde{M}+A)^{-1}$, is dense. We will thus make an approximation that will allow us to leverage faster DCFP solvers.

Two-step algorithm We make the assumption that in the solid and liquid zones, the Newtonian viscosity is small w.r.t. the frictional contact forces; that it, as we are mainly considering gravity-driven flows, that $\mathrm{Re} := \frac{\rho g^{\frac{1}{2}} L^{\frac{3}{2}}}{\eta}$ is high. We then proceed to solve an approximated version of (7.21) in two steps.

We first solve the unconstrained momentum balance (7.22) using a conjugate-gradient algorithm,

$$(\tilde{M}+A)\underline{\boldsymbol{u}}^* = \underline{\boldsymbol{l}}\,. \tag{7.22}$$

The frictional response of the material is then computed by neglecting the effect of the change in Newtonian stress due to the addition of contact stresses. That is, we write $\underline{\boldsymbol{u}} = \underline{\boldsymbol{u}}^* + \Delta\underline{\boldsymbol{u}}$, and solve a new DCFP:

Find $\Delta\underline{\boldsymbol{u}}$, $\underline{\boldsymbol{\lambda}}$, $\underline{\boldsymbol{\gamma}}$ s.t.

$$\begin{cases} \tilde{M}\Delta\underline{\boldsymbol{u}} = B^{\intercal}\underline{\boldsymbol{\lambda}} \\ \underline{\boldsymbol{\gamma}} = B\Delta\underline{\boldsymbol{u}} + B\underline{\boldsymbol{u}}^{*} + \underline{\boldsymbol{k}} \\ (\underline{\boldsymbol{\gamma}}_{[i]}, \underline{\boldsymbol{\lambda}}_{[i]}) \in \mathscr{D}\mathscr{P}(\mu) \qquad \forall 1 \leq i \leq n. \end{cases} \tag{7.23}$$

Doing so, we fully preserve the impact of the Newtonian viscosity in the gaseous regime, while neglecting its effect within zones that are dominated by static friction. Our DCFP's stiffness matrix is now diagonal, positive-definite and therefore trivial to inverse. This positively impacts the performance in two ways:

- We get a significant speedup using the solvers of Section 6.4, as the linear system solve at each iteration of the inner problem is now replaced with a single multiplication by a diagonal matrix.
- The Delassus operator of (7.23) is now sparse and easy to assemble, which opens the way to a wider range of algorithms.

In practice, we leverage our Gauss–Seidel algorithm from Chapter 4. Note though that in 3D, as the dimension of the constraint is 6, the analytical polynomial-based local solver cannot be used. However, we found that the Fischer–Burmeister-based solver worked well enough for the hybrid strategy not to be required. We also found the matrix-free variant of the algorithm to be more efficient than the original one, especially for $\mathbb{Q}_{1d}$ and particle-based stresses where the number of constraints is much higher than the number of velocity degrees of freedom. Attempting to assemble the Delassus operator with such discretization strategies could also quickly overwhelm our amount of available memory, even for relatively small test cases.

7.3.5 Overview of a time-step

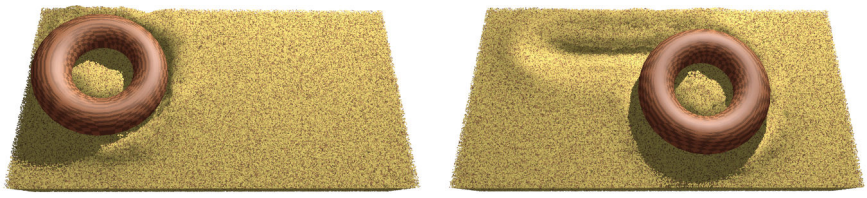
The full MPM algorithm can be summarized as follows:

1. Compute the lumped mass matrix $\tilde{M}$ with Eq (7.16);
2. Recover nodal velocities $\mathbf{u}^{p\mapsto g}$ from System (7.17);
3. Use Section 6.3 to assemble the matrices A and B and vectors $\underline{\boldsymbol{l}}$ and $\underline{\boldsymbol{k}}$ corresponding to the forms given in Section 7.3.1;
4. Solve for the unconstrained velocity $\mathbf{u}^{*}$ with Eq (7.22);
5. Solve DCFP (7.23) and get the total velocity $\mathbf{u} = \mathbf{u}^{*} + \Delta\mathbf{u}$;
6. Update particles frames with Eq (7.20), then split or merge them according to Section 7.3.2;
7. Compute the velocity gradient matrix $\boldsymbol{c}_p$ using Eq (7.19);
8. Update particle positions and velocities as per Eq (5.17, 5.18);
9. Proceed to next time-step.

The quantities that need to be stored on particles are therefore V_p, $\boldsymbol{x}_p$, $\boldsymbol{v}_p$, $\boldsymbol{c}_p$ and F_p. We also store the stress field $\boldsymbol{\lambda}$ between timesteps, in order to warm-start the DCFP solver.

7.4 Extensions

While collapsing granular columns may be deemed interesting in themselves, Computer Graphics applications often require the simulated material to interact with other external objects, such as an animated character. In this section, we present two extensions of our framework that could potentially allow artists to drive the simulation, or experiment with different look-and-feels by tuning the anisotropy of the flow. We insist on the fact that these "hacks" are designed to allow the visual enrichment of the granular flow, and have not been validated against any physical experiment.



(a) $\mu_{RB} = 0$ **(b)** $\mu_{RB} > 0$

Figure 7.12: *A slightly tilted wheel, on which a constant torque is applied, is dropped on a sandy ground.* ***(a)*** *Without wheel-sand friction, the wheel falls down immediately.* ***(b)*** *Otherwise, it manages to roll on for some distance.*

7.4.1 Rigid body coupling and frictional boundaries

In the following, we describe the modifications that have to be made to the variational formulation from Property 7.1 and the DCFP (7.23) to perform two-way coupling with rigid bodies. Frictional boundary conditions will then be imposed through the means of rigid bodies with infinite inertia.

Hard boundary For the sake of simplicity, we consider here the case where the rigid body Ω_{RB} is defined by an exact boundary, $\text{Bd}\,\Omega_{\text{RB}}$. An heuristic derivation for the case of a diffuse interface is given in (Daviet and Bertails-Descoubes 2016a).

We consider an additional stress field inside the rigid body, $-\boldsymbol{\lambda}_{\text{RB}} \in T_{\text{RB}} \subset L_2(\Omega_{\text{RB}})^{s_d}$. The reaction force applied by the rigid body onto the granular material through the boundary is given by $\boldsymbol{r} = \boldsymbol{\lambda}_{\text{RB}} \boldsymbol{n}_{\text{RB}}$, with $\boldsymbol{n}_{\text{RB}}$ the normal to $\text{Bd}\,\Omega_{\text{RB}}$. Let $\hat{\mathbf{v}}$ denote the relative velocity between the granular material and the rigid body; we want $\boldsymbol{r}$ and $\hat{\mathbf{v}}$ to follow a Coulombic relationship. A convenient way to enforce this in our framework is to add the condition $(\boldsymbol{\lambda}_{\text{RB}}, \boldsymbol{\gamma}_{\text{RB}}) \in \mathscr{DP}(\mu_{\text{RB}})$, where $\boldsymbol{\gamma}_{\text{RB}} := \frac{1}{2}\left(\hat{\mathbf{v}}\boldsymbol{n}_{\text{RB}}^{\intercal} + \boldsymbol{n}_{\text{RB}}\hat{\mathbf{v}}^{\intercal}\right)$ and the coefficient μ_{RB} sets the intensity of friction between the granular material and the rigid body (see Figure 7.12). The rationale behind this constraint is exposed in Appendix C.2. This formulation with two stress fields allows us to separate the constraints associated to grains–grains contacts from those associated to grains–rigid-body contacts, and thus to set distinct friction coefficients, as illustrated in Figure 7.12.

Proposition 7.2. *Let $\boldsymbol{v}_{RB} \in \mathbb{R}^6$ be the generalized velocity vector of the rigid body, i.e., the concatenation of the linear and angular velocities. The coupled granular–rigid body system then satisfies the variational formulation:*

Find $\mathbf{u} \in V$, $(\boldsymbol{\lambda}, \boldsymbol{\gamma}) \in T$, $(\boldsymbol{\lambda}_{RB}, \boldsymbol{\gamma}_{RB}) \in T_{RB}$,

$$\begin{cases} m(\mathbf{u},\mathbf{v}) + a(\mathbf{u},\mathbf{v}) = b(\boldsymbol{\lambda},\mathbf{v}) + b_{RB}(\boldsymbol{\lambda}_{RB},\mathbf{v}) + l(\mathbf{v}) & \forall \mathbf{v} \in V \\ \boldsymbol{w}^{\intercal} M_{RB} \boldsymbol{v}_{RB} = \boldsymbol{w}^{T} \boldsymbol{f} + c_{RB}(\boldsymbol{w}, \boldsymbol{\lambda}_{RB}) & \forall \boldsymbol{w} \in \mathbb{R}^6 \\ s(\boldsymbol{\gamma},\mathbf{u}) = b(\boldsymbol{\tau},\mathbf{u}) + k(\boldsymbol{\tau}) & \forall \boldsymbol{\tau} \in T \\ s(\boldsymbol{\gamma}_{RB},\mathbf{u}) = b_{RB}(\boldsymbol{\tau},\mathbf{u}) + c_{RB}(\boldsymbol{\tau}, \boldsymbol{v}_{RB}) & \forall \boldsymbol{\tau} \in T_{RB} \\ (\boldsymbol{\gamma},\boldsymbol{\lambda}) \in \mathscr{DP}(\mu) & \\ (\boldsymbol{\gamma}_{RB},\boldsymbol{\lambda}_{RB}) \in \mathscr{DP}(\mu_{RB}) & \end{cases} \tag{7.24}$$

where M_{RB} and $\boldsymbol{f}$ are the inertia matrix and vector of external forces of the rigid body (see Section 2.1.1), and where we have introduced two new bilinear forms, $b_{RB} : T_{RB} \times V \to \mathbb{R}$ and

$c_{RB} : T_{RB} \times \mathbb{R}^6 \to \mathbb{R}$,

$$b_{RB}(\tau, \mathbf{u}) := \frac{1}{2} \int_{\mathrm{Bd}\,\Omega_{RB}} \left(\mathbf{u} \boldsymbol{n}_{RB}^\intercal + \boldsymbol{n}_{RB} \mathbf{u}^\intercal \right) : \tau$$
$$c_{RB}(\tau, \boldsymbol{v}_{RB}) := -\frac{1}{2} \int_{\mathrm{Bd}\,\Omega_{RB}} \left(\mathrm{J}(\boldsymbol{v}_{RB}) \boldsymbol{n}_{RB}^\intercal + \boldsymbol{n}_{RB} \mathrm{J}(\boldsymbol{v}_{RB})^\intercal \right) : \tau,$$

with $\mathrm{J}(\boldsymbol{v}_{RB})$ *the velocity field spawned by a rigid body with linear and angular velocities* $\boldsymbol{v}_{RB}$ *and center of mass* $\boldsymbol{x}_{RB}$,

$$\mathrm{J}(\boldsymbol{v}_{RB})(\boldsymbol{x}) = \mathrm{J}\begin{pmatrix} \boldsymbol{v}_{RB}^{lin} \\ \boldsymbol{v}_{RB}^{ang} \end{pmatrix}(\boldsymbol{x}) = \boldsymbol{v}_{RB}^{lin} + \boldsymbol{v}_{RB}^{ang} \wedge (\boldsymbol{x} - \boldsymbol{x}_{RB}).$$

Discretizing System (7.24) once again yields a DCFP, which can be solved as usual.

Proof. By definition $\hat{\mathbf{v}} = \mathbf{u} - \mathrm{J}(\boldsymbol{v}_{\mathrm{RB}})$, so the weak definition of $\boldsymbol{\gamma}_{\mathrm{RB}}$ as per the fourth line of (7.24) is consistent.

We will now justify that the terms $b_{\mathrm{RB}}(\boldsymbol{\lambda}_{\mathrm{RB}}, \mathbf{v})$ and $c_{\mathrm{RB}}(\lambda_r b, \boldsymbol{w})$ added to the momentum conservation equations of the granular material and rigid body (i.e., the first two lines of (7.24)) do correspond to the interaction between the two media.

We recall the identity, for any symmetric tensor $\boldsymbol{\tau} \in S_d$, and for $(\boldsymbol{v}, \boldsymbol{n}) \in \mathbb{R}^d \times \mathbb{R}^d$,

$$\boldsymbol{v}^\intercal \boldsymbol{\tau} \boldsymbol{n} = \sum_i \boldsymbol{v}_i \sum_j \boldsymbol{\tau}_{ij} \boldsymbol{n}_j = \sum_{i,j} \boldsymbol{v}_i \boldsymbol{n}_j \boldsymbol{\tau}_{i,j} = \boldsymbol{\tau} : \boldsymbol{v} \otimes \boldsymbol{n} = \frac{1}{2} \boldsymbol{\tau} : (\boldsymbol{v} \otimes \boldsymbol{n} + \boldsymbol{n} \otimes \boldsymbol{v}). \tag{7.25}$$

The force $\boldsymbol{f}^C$ and torque $\boldsymbol{L}^C$ applied onto the rigid body by the granular material are given by (Batty et al. 2007):

$$\boldsymbol{f}^C = \int_{\Omega_{\mathrm{RB}}} \nabla \cdot [-\boldsymbol{\lambda}_{\mathrm{RB}}] \qquad\qquad \boldsymbol{L}^C = \int_{\Omega_{\mathrm{RB}}} (\boldsymbol{x} - \boldsymbol{x}_{\mathrm{RB}}) \wedge \nabla \cdot [-\boldsymbol{\lambda}_{\mathrm{RB}}].$$

For any $\boldsymbol{r} \in \mathbb{R}^3$, there holds

$$\begin{aligned}
\boldsymbol{r}^T \boldsymbol{f}^C &= \boldsymbol{r}^T \int_{\Omega_{\mathrm{RB}}} \nabla \cdot [-\boldsymbol{\lambda}_{\mathrm{RB}}] = -\int_{\mathrm{Bd}\,\Omega_{\mathrm{RB}}} \boldsymbol{r}^T \boldsymbol{\lambda}_{\mathrm{RB}} \boldsymbol{n}_{\mathrm{RB}} \\
&= -\int_{\mathrm{Bd}\,\Omega_{\mathrm{RB}}} \frac{1}{2} \boldsymbol{\lambda}_{\mathrm{RB}} : (\boldsymbol{n}_{\mathrm{RB}} \otimes \boldsymbol{r} + \boldsymbol{r} \otimes \boldsymbol{n}_{\mathrm{RB}}) && \text{using (7.25)} \\
\boldsymbol{r}^T \boldsymbol{L}^C &= \boldsymbol{r}^T \int_{\Omega_{\mathrm{RB}}} (\boldsymbol{x} - \boldsymbol{x}_{\mathrm{RB}}) \wedge \nabla \cdot [-\boldsymbol{\lambda}_{\mathrm{RB}}] = -\int_{\Omega_{\mathrm{RB}}} \boldsymbol{r}^T [\boldsymbol{x} - \boldsymbol{x}_{\mathrm{RB}}]_\wedge \nabla \cdot \boldsymbol{\lambda}_{\mathrm{RB}} \\
&= \int_{\Omega_{\mathrm{RB}}} \left([\boldsymbol{x} - \boldsymbol{x}_{\mathrm{RB}}]_\wedge \boldsymbol{r} \right)^\intercal \nabla \cdot \boldsymbol{\lambda}_{\mathrm{RB}} \\
&= \int_{\mathrm{Bd}\,\Omega_{\mathrm{RB}}} \left([\boldsymbol{x} - \boldsymbol{x}_{\mathrm{RB}}]_\wedge \boldsymbol{r} \right)^\intercal \boldsymbol{\lambda}_{\mathrm{RB}} \boldsymbol{n}_{\mathrm{RB}} - \underbrace{\int_{\Omega_{\mathrm{RB}}} \mathrm{D}([\boldsymbol{x} - \boldsymbol{x}_{\mathrm{RB}}]_\wedge \boldsymbol{r}) : \boldsymbol{\lambda}_{\mathrm{RB}}}_{=0} \\
&= \int_{\mathrm{Bd}\,\Omega_{\mathrm{RB}}} \frac{1}{2} \boldsymbol{\lambda}_{\mathrm{RB}} : \left(\boldsymbol{n}_{\mathrm{RB}} \otimes \left([\boldsymbol{x} - \boldsymbol{x}_{\mathrm{RB}}]_\wedge \boldsymbol{r} \right) + \left([\boldsymbol{x} - \boldsymbol{x}_{\mathrm{RB}}]_\wedge \boldsymbol{r} \right) \otimes \boldsymbol{n}_{\mathrm{RB}} \right) && \text{using (7.25)} \\
&= -\int_{\mathrm{Bd}\,\Omega_{\mathrm{RB}}} \frac{1}{2} \boldsymbol{\lambda}_{\mathrm{RB}} : (\boldsymbol{n}_{\mathrm{RB}} \otimes (\boldsymbol{r} \wedge [\boldsymbol{x} - \boldsymbol{x}_{\mathrm{RB}}]) + (\boldsymbol{r} \wedge [\boldsymbol{x} - \boldsymbol{x}_{\mathrm{RB}}]) \otimes \boldsymbol{n}_{\mathrm{RB}}),
\end{aligned}$$

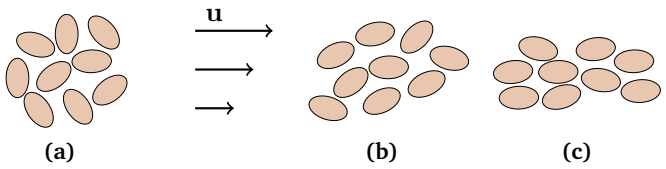


Figure 7.13: *Rationale for our handling of anisotropy. (a) (b) A shear flow induces a privileged orientation. (c) Aligned particles yield anisotropic macroscopic friction: reduced in the horizontal direction, more intense in the vertical one.*

which means, $\forall \boldsymbol{w} \in \mathbb{R}^6$,

$$\boldsymbol{w}^\intercal \begin{pmatrix} \boldsymbol{f}^C \\ \boldsymbol{L}^C \end{pmatrix} = c_{\mathrm{RB}}(\boldsymbol{\lambda}_{\mathrm{RB}}, \boldsymbol{w}),$$

and hence justifies the second line of (7.24).

Concerning the momentum conservation of the granular material, using (7.25),

$$b(\boldsymbol{\lambda}_{\mathrm{RB}}, \mathbf{v}) = \int_{\mathrm{Bd}\,\Omega_{\mathrm{RB}}} \langle \boldsymbol{\lambda}_{\mathrm{RB}} \boldsymbol{n}_{\mathrm{RB}}, \mathbf{v} \rangle,$$

which does correspond to the force applied by the rigid body onto the granular material through the boundary. □

7.4.2 Anisotropy

Many granular materials are composed of anisotropic grains that are thinner along one direction. This is typically the case for corn flakes, or, more common in the Computer Graphics imaginary, piles of gold coins (see Figure 7.23). It is noteworthy that anisotropy at the grain scale does play a role on the collective granular behavior. Indeed, while there is no reason to favor any particular direction when the grains are randomly oriented, when all their normals are aligned the macroscopic friction becomes anisotropic: it has much less dissipative effect when grains are sliding on top of each other, rather than when their relative displacement is along their common normal (see Figure 7.13 for an illustration).

However, remember that each one of our particles does not represent a single grain, but a collection of them, and thus it may contain different orientations. Rather than mapping a normal to each particle, we should instead store a probability distribution function (PDF) $\psi(\boldsymbol{n})$. For efficiency purposes we store only its second moment, the symmetric tensor $\boldsymbol{\nu}_2 = \int_{S^2} \boldsymbol{n}\boldsymbol{n}^\intercal \psi(\boldsymbol{n}) \mathrm{d}\boldsymbol{n}$. In the following, we propose a way to model the influence of $\boldsymbol{\nu}_2$ on our $\mathscr{D}\mathscr{P}(\mu)$ rheology, then construct an heuristic evolution equation for this tensor.

Modification of $\mathscr{D}\mathscr{P}(\mu)$ Anisotropy can be included in the rheology by simply replacing the norm $|\cdot|$ with the norm $|\cdot|_{\mathscr{A}}$ associated to the scalar product $\langle \cdot, \cdot \rangle_{\mathscr{A}} := \frac{1}{2}(\cdot : \mathscr{A}) : (\mathscr{A} : \cdot)$, where $\mathscr{A}$ is a fourth-order symmetric tensor. In order for the maximum dissipation principle to remain satisfied, $\boldsymbol{\gamma}$ should also be replaced with $\mathscr{A} : \boldsymbol{\gamma}$ on the first line of Equation (7.8).

We choose to construct $\mathscr{A}$ such as $\mathscr{A} : \boldsymbol{\tau} = N\boldsymbol{\tau}N$, i.e., $\mathscr{A}_{ijk\ell} = N_{ik}N_{\ell j}$, where the symmetric tensor N is deduced from the normal orientation tensor $\boldsymbol{\nu}_2$ following $N^{-1} = (1-\alpha)\mathbb{I} + d\alpha\boldsymbol{\nu}_2$, where $0 \leq \alpha \leq 1$ is a dimensionless coefficient parameterizing the amount of anisotropy in the frictional law. Using this formula, the effective friction coefficient will remain equal to μ for isotropic orientations (the d eigenvalues of $\boldsymbol{\nu}_2$ equal to $\frac{1}{d}$). If all the normals are oriented in the same direction (a single non-zero eigenvalue equal to 1), the effective friction coefficient will be $(1+(d-1)\alpha)^2\mu$ in the normal direction, and $(1-\alpha)^2\mu$ in the tangential ones.

Instead of modifying our friction solver, we can introduce the linear operator $\underline{\boldsymbol{N}} : \underline{\boldsymbol{\lambda}} \mapsto \frac{\mathrm{Tr}\boldsymbol{\lambda}}{d}\mathbb{I} + N\,\mathrm{Dev}\,\boldsymbol{\lambda}N$, make the changes of variable $\underline{\tilde{\boldsymbol{\lambda}}} = \underline{\boldsymbol{N}}\underline{\boldsymbol{\lambda}}$, $\underline{\tilde{\gamma}} = \underline{\boldsymbol{N}}\underline{\gamma}$ and replace the DCFP (7.23) with

$$\begin{aligned} \tilde{M}\Delta\underline{\boldsymbol{u}} &= B^{\intercal}\underline{\boldsymbol{N}}^{-1}\underline{\tilde{\boldsymbol{\lambda}}} \\ \underline{\tilde{\gamma}} &= \underline{\boldsymbol{N}}^{-1}B\Delta\underline{\boldsymbol{u}} + \underline{\boldsymbol{N}}^{-1}\left(B\underline{\boldsymbol{u}}^{*} + \underline{\boldsymbol{k}}\right) \\ (\underline{\gamma}_{[i]}, \underline{\boldsymbol{\lambda}}_{[i]}) &\in \mathscr{D}\mathscr{P}(\mu) \qquad\qquad \forall 1 \leq i \leq n. \end{aligned}$$

It now remains to describe how the second moment of the orientation normal PDF, $\boldsymbol{\nu}_2$, changes in time.

Evolution of $\boldsymbol{\nu}_2$ The evolution of the orientation of rigid ellipsoids in a shearing Newtonian flow has been described by Jeffery (1922). Integrating it over the unit sphere of all possible orientations (see e.g., Folgar and Tucker 1984) yields the evolution of the second moment tensor $\boldsymbol{\pi}_2$ of the orientation PDF,

$$\frac{D\boldsymbol{\pi}_2}{Dt} = \mathrm{W}(\mathbf{u})\boldsymbol{\pi}_2 + \boldsymbol{\pi}_2\,\mathrm{W}(\mathbf{u}) + \ell(\mathrm{D}(\mathbf{u})\boldsymbol{\pi}_2 + \boldsymbol{\pi}_2\,\mathrm{D}(\mathbf{u}) - 2\,\mathrm{D}(\mathbf{u}) : \boldsymbol{\pi}_4) \tag{7.26}$$

where $0 \leq \ell \leq 1$ is a coefficient describing the elongation of the ellipsoid, $\ell = \frac{L/W-1}{L/W+1}$, and $\boldsymbol{\pi}_4$ the fourth moment of the orientation PDF. The parameter ℓ affects the tendency of the ellipsoids to align with the flow. Note that this evolution equation does not directly apply to our problem:

- Jeffery's equation consider the velocity of a surrounding Newtonian fluid. In our case, the velocity field $\mathbf{u}$ is that of the granular matter itself.
- This model is only valid for dilute suspensions. Many authors have postulated laws for extending it to higher particle concentrations (e.g., Folgar and Tucker (1984) added an additional dissipative term modeling random collisions between particles in the semi-concentrated regime).

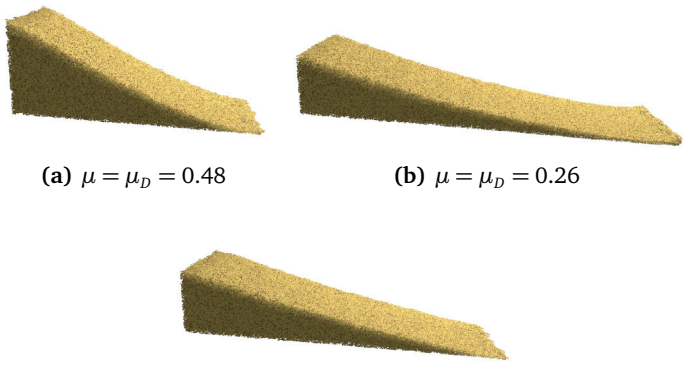
We will nonetheless take inspiration from this evolution equation and propose the following algorithm for the evolution of our normal orientation tensor $\boldsymbol{\nu}_2$, which yields good enough results for our purposes — for high values of ℓ, flat particles in a shearing flow do tend to become parallel to each other, which corresponds to the expected behavior (see Figures 7.13 and 7.15).

1. Deduce $\boldsymbol{\pi}_2(t)$ from $\boldsymbol{\nu}_2(t)$ as $\boldsymbol{\pi}_2 = \frac{1}{d-1}P(\mathbb{I} - D)P^{\intercal}$, where P and D are given by the eigen decomposition of $\boldsymbol{\nu}_2$.
2. Explicitly compute $\boldsymbol{\pi}_2(t + \Delta_t)$ using Equation (7.26) and the quadratic approximation of the $\boldsymbol{\pi}_4$ tensor, $\boldsymbol{\pi}_4 \sim \boldsymbol{\pi}_2 \otimes \boldsymbol{\pi}_2$ — the outer tensor product of $\boldsymbol{\pi}_2$ with itself.
3. Deduce $\boldsymbol{\nu}_2(t + \Delta_t)$ from $\boldsymbol{\pi}_2(t + \Delta_t)$, and normalize it using $\boldsymbol{\nu}_2 \leftarrow \frac{\boldsymbol{\nu}_2}{\|\boldsymbol{\nu}_2\|_1}$.

7.5 Results

We used this hybrid framework to perform 3D simulations of granular materials. Rendering was done following the method of Narain, Golas, et al. (2010), yielding several samples for each simulation particle, but using rasterization instead of ray-tracing. All simulation code, including the finite-element discretization, relied on a purposely-built library, that has since then been released as open-source[2]. Unless otherwise mentioned, trilinear shape functions were used for the stresses. The DCFP were solved using the `bogus` library. As usual, all our simulations and benchmarks were run on Intel® Xeon® quad-core workstations.

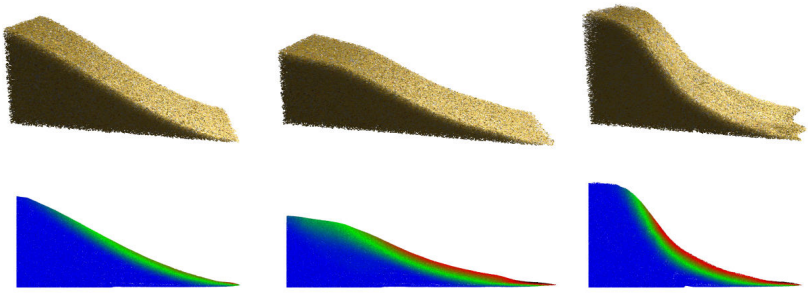
[2] http://bipop.inrialpes.fr/~gdaviet/code/sand6



(a) $\mu = \mu_D = 0.48$ **(b)** $\mu = \mu_D = 0.26$

(c) $\mu = 0.26$, $\mu_D = 0.48$

Figure 7.14: *Collapsed granular columns with different friction coefficients. Remark that the dynamic friction coefficient μ_D of the $\mu(I)$ rheology influences the run-out length, but not the rest angle.*



(a) *isotropic friction* **(b)** $\alpha = 0.8$, *with random initial particle normals* **(c)** $\alpha = 0.8$, *with upward initial normals*

Figure 7.15: *Snapshot and corresponding velocity field of the anisotropic column collapse with $\mu = 0.48$ at $t = 2.72$.*

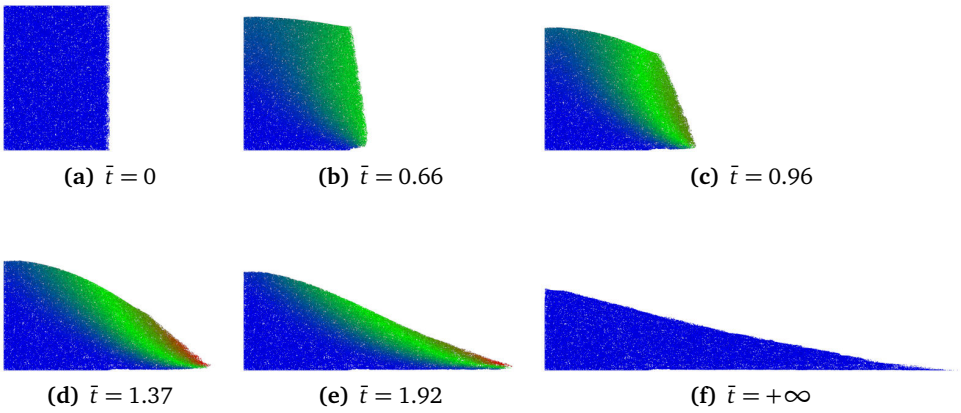
7.5.1 Model problems

Here we study the influence of several parameters of our model on the collapse of a 3D rectangular column — the experiment setting as for the $\mathbb{P}_0$ discretization in Section 7.2.3, though with finite (and constant) depth. We refer the reader to the accompanying video [3] for a more comprehensive view of their dynamical effects.

Influence of friction coefficients Figure 7.14 depicts the final, stable states following the collapse for the constant μ and $\mu(I)$ rheologies. While μ is directly related to the slope of the final granular heap (higher μ implies steeper slope), $(\mu_D - \mu)$ has an effect on the dynamic regime, and therefore on the horizontal spread of the heap. Cranking up the Newtonian viscosity parameter η also reduces this spread, but gives a muddier look to the simulation.

Influence of anisotropy Figure 7.15 compares the profiles and velocities of the column collapse for different anisotropy settings. The anisotropic collapse with random initial orientations yields a longer run-out, as the particles align with the flow and minimize friction. The distinct frictional responses of the yielded and unyielded zones are even more visible when initial orientations are uniform.

[3] http://bipop.inrialpes.fr/~gdaviet/files/mpm/mpmGranular.mp4



(a) $\bar{t} = 0$ **(b)** $\bar{t} = 0.66$ **(c)** $\bar{t} = 0.96$

(d) $\bar{t} = 1.37$ **(e)** $\bar{t} = 1.92$ **(f)** $\bar{t} = +\infty$

Figure 7.16: *Orthographic views of a collapsing column with aspect-ratio* 1.4 *at selected instants in time, to be compared with Lagrée et al. 2011, Figure 9. Colors denote particles velocities (blue slowest, black fastest), and can be compared with a similar visualization in Ionescu et al. 2015, Figure 2.*

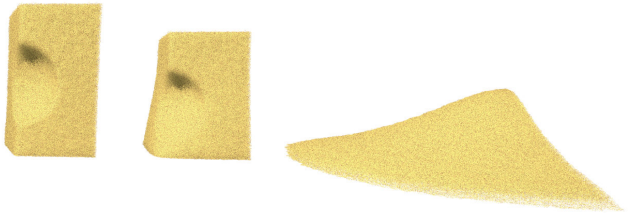


Figure 7.17: *Revisiting Y. Zhu and Bridson 2005's column collapse.*

Comparisons Continuum simulations using the $\mu(I)$ rheology were recently validated against DEM simulations (Lagrée et al. 2011) and real experiments (Ionescu et al. 2015) of a 2D granular column collapse. We have reproduced the experiment of Lagrée et al. (2011, Figure 9). The friction parameters $\mu = 0.26$, $\mu_D = 0.48$ of our 3D simulation were chosen so as to match those from (Lagrée et al. 2011) using the midpoint Drucker-Prager surface (see Section 0.3.2). Figure 7.16 shows that we retrieve the correct profiles throughout time. While our final relative height, $H_\infty/H_0 = 0.56$, quantitatively matches the 2D experiments, our final dimensionless run-out length, $\frac{L_\infty - L_0}{L_0} \sim 2.98$, is slightly slower.

Figure 7.17 depicts representative frames of our simulation of the sand column collapse scenario introduced by Y. Zhu and Bridson (2005). Our results are comparable to those later obtained by Narain, Golas, et al. (2010), where the incompressibility constraint was relaxed. However, Figure 7.1 illustrates that replacing the Frobenius norm of the Drucker-Prager law with ℓ_∞, as done in (Narain, Golas, et al. 2010), yields anisotropic artifacts, as the effective friction coefficient then ranges from μ to $\sqrt{5}\mu$ depending on the flow direction.

Retrieving the empirical laws of silo discharge Focusing our hybrid method on the simulation of a 3D rectangular granular silo with round outlet (Figure 7.18), we attempt to retrieve the empirical laws that were captured by our 2D simulations, the constant discharge rate (Section 7.2.3) and the Beverloo scaling (Section 6.5.3).

Figure 1.16(a), shows that a high enough friction coefficient is necessary to observe the constant discharge rate, while Figure 1.16(b) demonstrates that we do retrieve the power $\frac{5}{2}$ scaling of the 3D Beverloo law, $Q = C(D-k)^{\frac{5}{2}}$ where C and k are properties of the material and

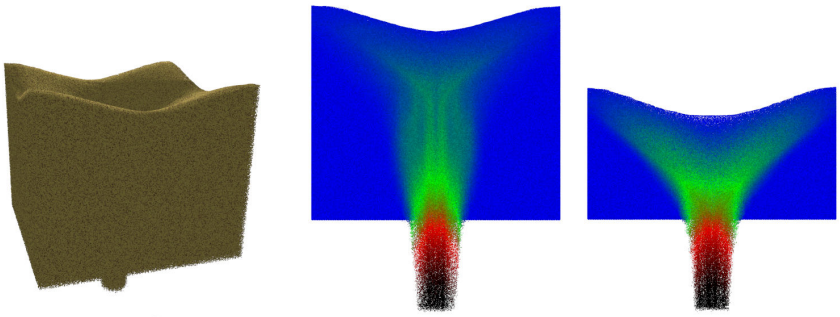
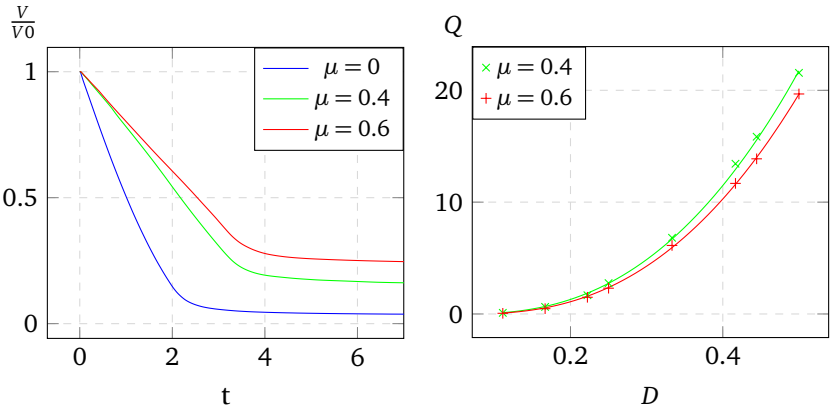


Figure 7.18: *Perspective view and orthographic projection of the particle velocities at different instants of the silo discharge.*



(a) *Normalized remaining volume over time for different friction coefficients* **(b)** *Simulated discharge rate vs outlet width (marks), and Beverloo fits (lines)*

Figure 7.19: *Constant discharge rate and Beverloo scaling for a 3D granular silo.*

silo geometry. However due to our relatively rough discretization, the resulting k corresponds to about two grid cells – instead of a few grain diameters in reported experiments.

7.5.2 Complex scenarios

Free-flowing material We simulated two scenarios in which sand is manipulated through scripted rigid-body motions. The first one, reproduced in Figure 7.20, features a small cylinder being dragged across the ground. We capture the formation of a typical mound at the front of the cylinder, and the permanent marks that are due to the frictional nature of the material. In the second scenario (Figure 7.21), a handful of sand is picked up before being let to flow freely, illustrating the transitions between the gaseous, liquid and solid regimes of the material.

Impact We ran two simulations reproducing impacts of a fully-coupled rigid sphere on granular media, with and without cohesion. While Narain, Golas, et al. (2010) were able to capture the dynamics of the impact of a two-inch tungsten ball on a granular bed[4], their simulation suffered from visible artifacts due to their linearization of the Drucker-Prager law. In contrast, our simulation of the same scenario yields a perfectly round crater (Figure 7.7), matching much more closely that of their reference video[5].

[4] https://youtube.com/watch?v=ZoZ0ZAzr6eg#t=90
[5] http://dsc.discovery.com/videos/time-warp-deep-impact

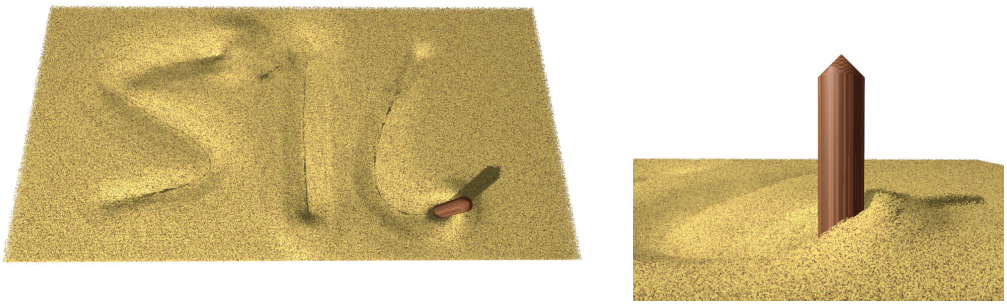


Figure 7.20: *Letters drawn by dragging a stick in the sand. Right: a typical mound grows at the front of the stick.*

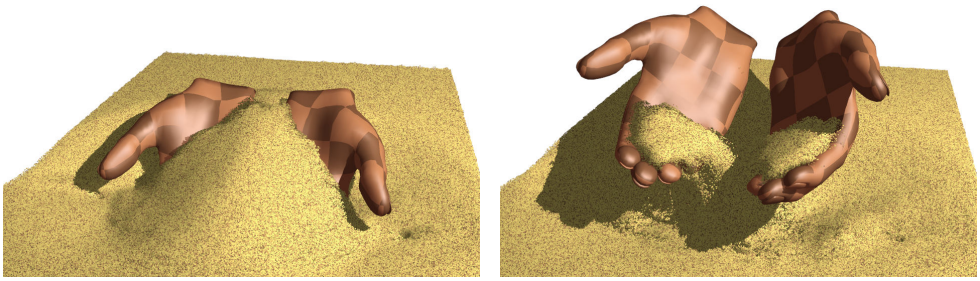
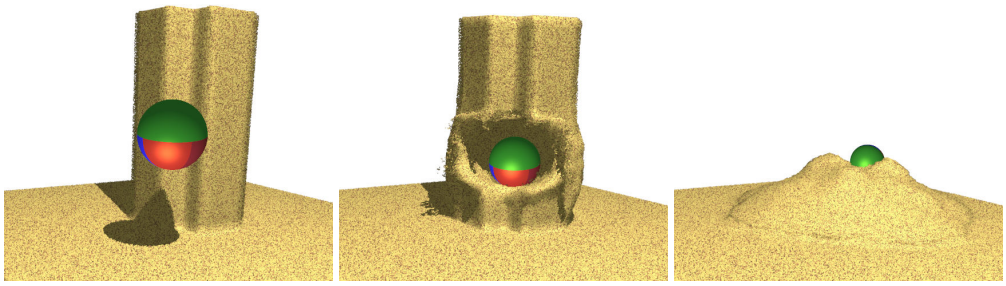
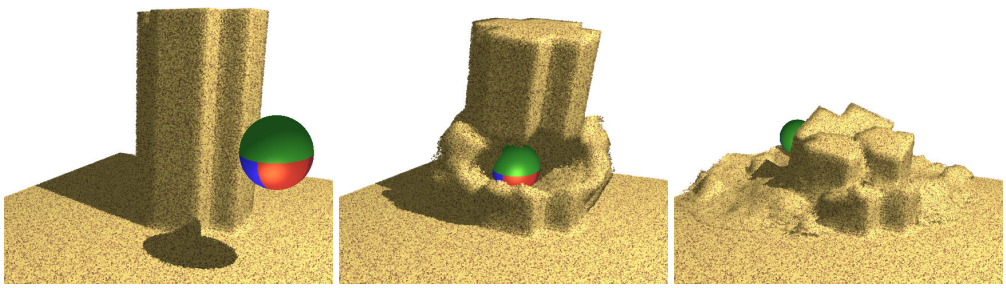


Figure 7.21: *Picking-up sand and letting it flow away.*



(a) *Trilinear stress shape functions and cohesion decay*



(b) *Particle-based stress shape functions without cohesion decay*

Figure 7.22: *A sphere impacting a sand tower initially standing up thanks to a high cohesion coefficient.*

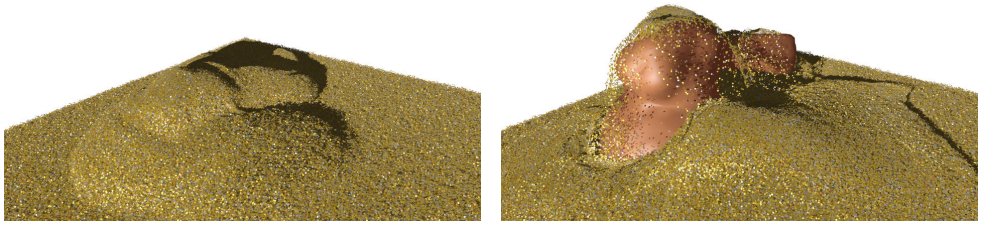


Figure 7.23: *Treasures made of gold coins are a typical example of highly anisotropic granular materials.*

Example	FPS	Grid	n_P [1]	n_N [1]	$\bar{t}$ [2]	$\%_b$	$\%_s$ [3]
Collapse	96	40×10×20	5.0 10^4	2755	1.47	19	72
ZB05	96	100×100×50	5.1 10^5	34112	23.5	21	71
Wheel	96	42×28×14	6.5 10^4	10152	9.62	4	92
Silo	96	36×36×72	1.3 10^6	49774	61.5	41	41
Writing	60	96×48×24	4.9 10^5	61864	48.4	18	65
Cohesion	240	80×80×40	3.3 10^5	42205	6.8	32	52
Digging	96	64×64×32	2.2 10^6	75283	32.3	23	65
Crater	600	75×75×50	8.3 10^5	110887	89.6	19	65
Treasure	96	70×70×50	3.3 10^6	121143	83.6	27	59

[1] Maximum numbers of particles (n_P) and active grid nodes (n_N)
[2] Average simulation time in seconds per rendering frame $\bar{t}$
[3] Percentage of time for **b**uilding and **s**olving the DCFP

Table 7.1: *Sizes and simulation time of our examples*

Cohesion As mentioned in Section 7.3.3, cohesion fits in very nicely with the continuous model, but creates tremendous discretization issues. When using trilinear shape functions for the stresses, we can only obtain a stable cohesive medium when the particles are aligned with the underlying grid. One solution to mitigate this limitation is to model debonding, for instance using a simple decay model linear in the norm of the strain rate $|\dot{\varepsilon}|$, $\frac{\mathrm{dc}}{\mathrm{d}t} = -\xi|\dot{\varepsilon}|\mathrm{c}$. Figure 7.22 reproduces the destruction of a granular tower initially standing thanks to cohesion using either this approach or particle-based stress shape functions. The latter option allows to retain much sharper features in the final state.

Anisotropy Scenes featuring large piles of gold coins are common in movies; we have simulated the Stanford bunny waking up under a large heap of coins, triggering anisotropic avalanches (see Figure 7.23).

7.5.3 Performance

Simulation timings Table 7.1 provides the sizes and simulation timings for all our examples. The target framerate (FPS) is dictated by the desired playback speed of our accompanying video — simulation frames were then subdivided using either a fixed number of substeps, or an adaptive criterion based on the fastest particle speed to grid cell size ratio. The tolerance for the DCFP solutions was set to 10^{-3} times the typical stress $\rho g L$, and was typically reached within 25 to 250 iterations of the Gauss–Seidel algorithm warm-started with the solution of the previous timestep.

The computational bottleneck of our method lies in the DCFP solving. Note that with trilinear stress basis functions, this cost increases with the number of active grid nodes, and has very little dependence on the number of simulated particles. We can therefore use a high numbers of particles per cell, such as in the Silo simulation, without incurring too much overhead.

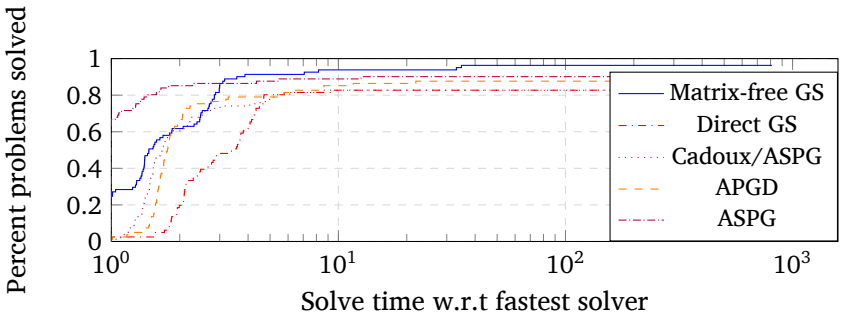


Figure 7.24: *Performance profiles for a variety of solvers. The line $x = 1$ gives the solvers that were the most often the fastest, while $y = 1$ shows the most robust ones.*

Comparisons between Coulomb friction solvers As mentioned previously, we use the open-source bogus library, which implements a few algorithms for solving DCFP, either tackling it directly or decomposing it as a sequence of optimization problems using the Cadoux algorithm (Section 2.3.2). In both cases, one may use our Gauss–Seidel (GS) algorithm from Chapter 4, or a few variants of the Projected-Gradient (PG) algorithm, including the APGD algorithm (Mazhar et al. 2015) and a line-search-free implementation of the ASPG method (Algorithm B.2).

Figure 7.24 shows the performance profiles for those solvers on a variety of problems extracted from our simulations — that is, the percentage of problems solved under a certain multiple of the time taken by the best-performing solver for each problem. To avoid favoring a given algorithm, we used the Alart-Curnier complementarity function to evaluate all the residuals equally. We found that the matrix-free versions of both PG and GS algorithms performed better than when explicitly assembling the Delassus operator, especially when the number of constraints (i.e., the number of stress degrees of freedom) was much larger than the velocity degrees of freedom. Note that the improved robustness of the matrix-free Gauss–Seidel shown in Figure 7.24 simply comes from the fact that for the bigger problems, the Delassus operator did not fit into main memory. We also found that using the Cadoux algorithm did not improve efficiency nor robustness on our benchmark problems. Overall, we found the matrix-free GS to perform the best on our quad-core setup and for our chosen tolerance. Note however that PG algorithms should theoretically scale more easily to a higher number of processors.

7.5.4 Limitations

While our hybrid method is applicable to various challenging scenarios, it still suffers from a number of limitations. Apart from the those inherent to the use of a continuum model — which we will discuss later — the choice of the discrete function space remain the main limitation of the method, and compromises have to be made. Trilinear stress shape functions are computationally efficient, but may lead to the creation of artifacts. In particular, as mentioned in Section 7.3.3, cohesive materials have to be aligned with the underlying grid in order to remain stable, which drastically reduces their relevance. Moreover, disturbing artifacts may happen when visually disjoint clusters of particles in neighboring cells react together. Particle-based stresses mitigate these artifacts, but may lead to a much higher number of constraints and potential kinematic locking. Discontinuous trilinear stresses are a good compromise between the two approaches, but they are still expensive to solve. Moreover, enforcing the rheology to be satisfied only at a few discrete points in space may induce an overall loss of volume. A volume correction step may be applied, as in (Narain, Golas, et al. 2010), at the risk of introducing artificial energy in the system.

7.6 Discussion

In this chapter, we presented an extension of the numerical method of Chapter 6 that is applicable to general granular flows with varying volume fraction of grains. We saw that in order to obtain a symmetric system, the maximal volume fraction constraint has to be discretized using the Lagrangian point of view — i.e., looking at positions that are moving with the grains. The main difficulty in our approach consists in finding appropriate discretization spaces for the velocity, stress and volume fraction fields; we proposed two different choices, piecewise-constant discretization or the Material Point Method with low-order velocities and stresses. However, both of those choices suffer from limitations; finding better discretization spaces remain one of our priority for future work. Despite this, our method was still able to capture qualitative characteristics of granular flows, and to reduce the visual artifacts of previous approaches such as (Narain, Golas, et al. 2010) at a roughly similar computational cost. Moreover, in contrast to the cited approach, our numerical method derives from a proper, consistent, variational formulation rather than an ad-hoc sequence of corrective steps. This will allow us to accommodate the more complex case of diphasic flows in the next chapter.

The continuous model itself is also subject to several limitations; due to our continuum approximation of granulars, the range of possible simulations is restricted to homogeneous materials where all grains share common features, such as their size. Our method is thus not appropriate for simulating polydisperse media with various grain shapes. Moreover, being a macroscopic law, our $\mathscr{DP}(\mu)$ rheology cannot model specific arrangements of grains, such as the formation of arches clogging the silo outlet. Our definition of a constant maximal volume fraction is also flawed; it is well known that different packings are possible over time for a given granular medium, exhibiting different volume fractions. Moreover, Roux and Radjai (1998) state that the packing volume fraction has an influence on the dilatancy angle; our model is unable to capture this phenomenon. Finally, our equations are based on an inelastic impact assumption, while a non-zero restitution coefficient might be necessary to properly model some classes of granular flows.

A last limitation, which we are going to address in the next chapter, is that we completely ignored the presence of a fluid around the grains. While the influence of air on large-enough grains is negligible, this hypothesis prevents us from tackling interesting scenarios, such as immersed avalanches.

8 *Granular flows inside a fluid*

This chapter is dedicated to the simulation of a granular material inside a Newtonian fluid. In contrast with the previous chapter, the surrounding fluid is no longer considered to be much lighter than the grains. Our main objective is to capture the qualitative influence of the surrounding fluid on the dynamics of the granular material; for instance, looking at the dynamics of an ash cloud, or at how water will affect the avalanching behavior of a column of sand. However, we still want said avalanche to eventually come to rest. For this reason, we still assume the existence of a critical volume fraction of grains $\phi_{\max}$ such that when ϕ reaches $\phi_{\max}$, the grains interact together through dry frictional contact, and obey the non-associated Drucker–Prager flow rule. We also suppose that no mass transfer occurs between the two phases (i.e., no chemical reaction).

Even more than in the dry case, immersed granular materials can exhibit different regimes, depending for instance on whether the inertia of the grains or that of the fluid is preponderant (Topin, Monerie, et al. 2012); we will attempt to capture this behavior in our continuum simulations. As we will see, the ratio of the particle size w.r.t. the domain will be of utmost importance. To the best of our knowledge, the method presented in this chapter is the first to combine fully-coupled two-phase equations for immersed granular flows with implicit, nonsmooth treatment of the Drucker–Prager rheology. An interesting property of our method is that it will ultimately lead to equations very similar to that of the dry case discussed in Chapter 7. The main differences will be the presence of a second velocity field (the *fluctuation*, modeling the velocity difference between the two phases), and linear constraints enforcing the conservation of the local volume of matter.

Notations Quantities associated with the granular phase will be denoted with a "g" subscript (e.g., $\mathbf{u}_g$ will be the velocity field of the grains) while those associated with the surrounding fluid will be denoted with a "f". As in the previous chapters, ϕ will denote the grain volume fraction, and we will assume that the fluid occupies the entirety of the volume where there are no grains. That is, the density of the granular phase will be $\rho_g(\phi) = \rho_g\phi$, and the density of the fluid phase will be $\rho_f(\phi) = \rho_f(1-\phi)$. The total density of the mixture will thus be $\rho(\phi) = \rho_f(\phi)+\rho_g(\phi)$. We also define two velocities for the mixture: the mass-averaged velocity, $\mathbf{u}_{\mathrm{m}}$, such that $\rho(\phi)\mathbf{u}_{\mathrm{m}} = \rho_g(\phi)\mathbf{u}_g + \rho_f(\phi)\mathbf{u}_f$, and the volume-averaged velocity, $\mathbf{u}_{\mathrm{v}} := \phi\mathbf{u}_g + (1-\phi)\mathbf{u}_f$.

8.1 Related work

As the behavior of dry granular materials has yet to be fully understood, it is not surprising that no general model exists for immersed granular materials. However, they have been the subject of extensive studies in the second-half of the last century, in a variety of domains. Different models for different applications evolved largely in parallel, but cross-pollinated over time.

The study of the influence of dispersed grains in a Newtonian fluid goes as far as Einstein (1906), who postulated that the effective viscosity η_{eff} of a *slurry* — a dilute suspension of spherical particles — followed the law $\eta_{\mathrm{eff}}(\phi) = \eta_f(1+\frac{5}{2}\phi)$, where η_f is the viscosity of the pure Newtonian fluid, and ϕ the volume fraction of particles.

Several applications have motivated subsequent work on related problems. In rough chronological order, we can mention:

- Civil engineering and soil mechanics, with the work of Terzaghi (1936) on the reduction of the effective stress in a saturated porous material; later, with the poroelasticity theory of Biot (1955);
- Sedimentation and consolidation, of practical interest for the mining industry, with the seminal work of Kynch (1952);
- Fluidized granular beds, of special importance in chemical engineering, following Jackson (1963);
- Particulate gravity currents — that is, gravity-induced flows that mostly spread in the horizontal direction, such as avalanches (Parker et al. 1986);
- Sediment transport in river beds (Hanes and Bowen 1985);
- Finally, Computer Graphics (Lenaerts and Dutré 2009; Rungjiratananon et al. 2008).

To construct our numerical model, we will mainly follow the ideas of Jackson (2000). However, we propose below to walk through the different approaches that have been explored for the continuum simulation of grains in a Newtonian fluid.

8.1.1 Modeling

Kynch batch flux density Batch sedimentation study the gravity-induced settling process of initially dilute particles in a tank. Concha and Bürger (2002) relate the advances of sedimentation theory from the early works of Kynch (1952), who postulated that the evolution of the volume fraction of sediments in a horizontally homogeneous vertical tank obeyed a $1D$ equation,

$$\frac{\partial \phi}{\partial t} + \frac{\partial f_{\mathrm{BK}}(\phi)}{\partial z} = 0,$$

where f_{BK} is called the Kynch *batch flux density* function. As mass conservation states that $\frac{\partial \phi}{\partial t} = -\frac{\partial \phi \mathrm{u}_z}{\partial z}$, this is equivalent to saying that the vertical velocity depends only on the local volume fraction of sediments. An underlying assumption is that the volume fraction of grains is monotonically decreasing with altitude. Richardson and Zaki (1954) proposed an expression for the batch flux function, leading the vertical velocity to decrease with the volume fraction of fluid,

$$\mathrm{u}_z(\phi) = w_\infty (1-\phi)^\nu, \tag{8.1}$$

where w_∞ denotes the settling velocity of a single particle, and the exponent ν is generally observed to be greater than 3. Michaels and Bolger (1962) refined this expression to ensure that the vertical velocity vanishes at the bottom of the tank, where the particles have already reached a critical volume fraction $\phi_{\max}$ and cannot be compacted anymore. They use

$$\mathrm{u}_z(\phi) = w_\infty \left(1 - \frac{\phi}{\phi_{\max}}\right)^\nu.$$

Full two-phase models In parallel, Jackson (1963) started the study of fluidized granular beds by modeling this phenomenon as the interactions between two continua. However, he did not include any stress preventing compression of the granular phase, and his model was found to be unstable, disagreeing with observations. Anderson and Jackson (1967) derived the governing equations of the two continua using an averaging process. In particular, they demonstrated the necessity of including a generalized buoyancy contribution in the momentum transfer term between the two phases. They modeled the drag applied by the fluid on the particles using two terms: one term opposing the relative velocity of the two phases, determined from the sedimentation experiments of Richardson and Zaki (1954); and a second term opposing the acceleration between the two phases, a so-called virtual mass. Their work has ultimately led to the popular two-phase framework presented in (Jackson 2000).

Drew and Lahey (1979) proposed a general averaging framework for the derivation of thermodynamically admissible constitutive equations, and Drew (1983) proposed closures for common two-phase flows, specifying in particular many potential contributions to the interfacial momentum transfer term. Drew (1983) also suggested the use of an (unilaterally) incompressible model for the granular phase, stating that the particulate pressure should vanish while the maximum concentration is not reached.

Batchelor (1988) argues that sediment and granular bed fluidization are two facets of the same problem, and that both should be able to be described by a single model. He attributes the difference in the two communities approaches to the fact that "Fluidized beds have mostly been studied with larger particles, such that the Reynolds number of the flow about a particle is well above unity, whereas sedimentation processes in practice usually involve a liquid continuous phase and smaller particles for which the Reynolds number is small." Moreover, he argues that the work of Drew is hardly usable in practice; "One can go part way towards finding an equation of motion for one of the phases which formally resembles an equation for a continuum by taking an average over the volume occupied instantaneously by that phase in the manner described by Drew (1983), but his procedure achieves rigor at the cost of introducing the intractable problem of closure of averages of a complicated kind." Instead, assuming a one-dimensional mean velocity, he proposes a model whose parameters could all be measured from physical experiments. In particular, he proposes once again to deduce the drag force from sedimentation experiments, and models the inter-particle contacts as a diffusive phenomenon, with a force proportional to the gradient of the density of grains and a supplemental viscosity.

Harris and Crighton (1994) postulates the stress in the granular phase to be the sum of a Newtonian viscosity and a particulate "pressure" p^C growing to infinity when the volume fraction reaches $\phi_{\max}$, $\mathrm{p}^C := P_0 \phi \frac{1}{\phi - \phi_{\max}}$. The drag force is once again modeled using the sedimentation velocity given by Richardson and Zaki (1954). Bürger (2000) proposes different models for the viscosity of each phase, and uses a virtual mass and a batch flux density function to define the drag forces. Hsu et al. (2004) propose complex expressions for the drag — based on the Richardson and Zaki (1954) model — and inter-particle pressure — diverging for a volume fraction slightly above random close packing— targeted at the modeling of sediment transport.

Mixtures *Mixture* theories attempt to avoid fully modeling the two phases and their interactions through the use of higher-level closures. For instance, Frankel and Acrivos (1967) and Krieger (1972) propose to extend Einstein's law to denser concentrations of particles, making sure that the viscosity goes to infinity when the critical volume fraction of grains $\phi_{\max}$ is reached. Frankel and Acrivos (1967) write the effective mixture viscosity

$$\eta_{\text{eff}}(\phi) = \eta_f \frac{9}{8} \left(\left(\frac{\phi_{\max}}{\phi} \right)^{\frac{1}{3}} - 1 \right)^{-1}, \tag{8.2}$$

while Krieger (1972) suggests

$$\eta_{\text{eff}}(\phi) = \eta_f \left(1 - \frac{\phi}{\phi_{\max}} \right)^{-\frac{5}{2}\phi_{\max}}, \quad \nu > 0. \tag{8.3}$$

so that Einstein's law is retrieved in the dilute limit. Boyer et al. (2011) empirically attempt to unify the $\mu(I)$ rheology for dense and dry granular materials with mixture theory. They find an expression for the tangential stress which like that of Krieger (1972), diverges when ϕ reaches $\phi_{\max}$ and coincides with Einstein's viscosity in the dilute limit. The pressure-to-frictional-stress ratio is found to follow a $\mu(I_\nu)$ relationship, where I_ν is called the *viscous number*.

The mixture theory developed for gravity currents use different kinds of closures. A large portion of natural gravity currents involves phases with similar densities; in this case, many works rely on the Boussinesq approximation, which consists in considering the varying density solely in the buoyancy term (through the introduction of a "reduced" gravity). Ungarish (2009) considers this approximation to remain valid for density differences of up to 10% or

20%. However, neither granular avalanches in water, nor powder avalanches in the air, fall into this range. For higher density ratios, mixture theory (see Manninen et al. 1996 for a high-level introduction, or Bedford and Drumheller 1983 for a derivation from thermodynamical and averaging considerations) propose an alternative approximation of the multiphase continuum by considering closure equations for the diffusion velocity of the different phases, for instance following Fick's law (e.g., Etienne 2004). Recalling that $\mathbf{u}_\mathrm{m}$ denotes the mass-averaged velocity, $\mathbf{u}_\mathrm{m} := \frac{1}{\rho(\phi)}\left(\rho_g(\phi)\mathbf{u}_g + \rho_f(\phi)\mathbf{u}_f\right)$, the flux density $\mathbf{q}$ for each phase is defined such that

$$\frac{D_{\mathbf{u}_\mathrm{m}}\phi}{D_{\mathbf{u}_\mathrm{m}}t} + \phi\nabla\cdot\mathbf{u}_\mathrm{m} = -\nabla\cdot\left[\mathbf{q}_g\right]$$
$$\frac{D_{\mathbf{u}_\mathrm{m}}(1-\phi)}{D_{\mathbf{u}_\mathrm{m}}t} + (1-\phi)\nabla\cdot\mathbf{u}_\mathrm{m} = -\nabla\cdot\left[\mathbf{q}_f\right].$$

Expressing those flux densities using mass conservation would involve the velocities of individual phases velocities; $\mathbf{q}_g = (\mathbf{u}_g - \mathbf{u}_\mathrm{m})\phi$ and $\mathbf{q}_f = (\mathbf{u}_f - \mathbf{u}_\mathrm{m})(1-\phi)$. Fick's law instead postulates that the flux densities are proportional to the gradient of the volume fraction: $\mathbf{q}_g := q_g(\phi)\nabla\phi$, $\mathbf{q}_f := q_f(\phi)\nabla\phi$. Note that the rates q_f and q_g cannot be chosen independently, as total mass conservation requires $\rho_f q_f + \rho_g q_g = 0$. Remark that Fick's law amount to considering that the drift velocities $\mathbf{u}_g - \mathbf{u}_\mathrm{m}$ and $\mathbf{u}_f - \mathbf{u}_\mathrm{m}$ depend only on the volume fraction, which is reminiscent of Kynch's theory in a different, higher-dimensional setting. The evolution of the volume fraction field can then be expressed as an unsteady convection-diffusion equation,

$$\frac{D_{\mathbf{u}_\mathrm{m}}\phi}{D_{\mathbf{u}_\mathrm{m}}t} + \phi\nabla\cdot\mathbf{u}_\mathrm{m} + \nabla\cdot\left[q_f(\phi)\nabla\phi\right] = 0.$$

However, this diffusive model is not satisfying for our purposes, as we want to be able to simulate not only gravity currents but also sedimentation processes (such as sand falling at the bottom of a tank). As such, and since batch sedimentation theory is mostly concerned with 1D problems and is thus also too restrictive, our numerical model will be based on full two-phase equations.

8.1.2 Numerical simulations

These different modeling approaches have led to a variety of numerical methods for the simulation of immersed granular or powder flows. We mention below a few techniques that are representative of the diversity of strategies employed for this purpose, but the reader should keep in mind that this list is far from exhaustive.

Sedimentation and fluidization Andrews and O'Rourke (1996) extended the Particle-In-Cell hybrid method to the simulation of multiphase flow, using the particle interaction stress from (Harris and Crighton 1994). Apte et al. (2003) used a similar approach with a slightly different expression for the particulate phase pressure, using a non-unit exponent for the hyperbolic divergence of the stress when ϕ approaches ϕ_max. Chauchat and Médale (2010) model sediment transport in a laminar river bed by decomposing the flow into two (fixed) domains: the upper one where a mixture is considered, and the bottom one where the grains are assumed to be dense and a full two-phases continuum is considered. In the upper domain (the river), Einstein's viscosity for dilute suspensions is considered. The lower domain (the granular bed) is modeled following Jackson (2000), and the grains are subject to a pressure-dependent yield-stress. The volume fraction of grains is assumed to remain constant, and thus all flows are incompressible. The equations are then solved using FEM with regularization of the yield stress. Revil-Baudard and Chauchat (2013) study the transport of sediments by a turbulent fluid, using again a mixture approach for the upper, turbulent layer and a two-phase model for the bottom one. This granular layer is modeled using the $\mu(I_v)$ rheology from Boyer et al. (2011), and solved numerically

by regularizing of the yield stress. Chauchat, Guillou, et al. (2013) simulate 1D sedimentation processes by starting from the two-phase equations of Jackson (2000) with Frankel–Acrivos viscosity (8.2), and providing closures for the drag force and effective granular stress for cohesive and non-cohesive particles. They found their results to be in agreement with experiments in both cases.

Gravity currents Mixture theory has been especially popular for the numerical simulation of avalanches; Etienne (2004) and Birman et al. (2005) apply this approach to lock-exchange problems with high density ratios. Iverson (1997), and Denlinger and Iverson (2001) extend the depth-averaged Savage–Hutter model for dry granular flows to accommodate the presence of an interstitial fluid. Pitman and Le (2005) propose a new depth-averaged model by building upon the two-phase equations of Jackson (2000), and demonstrate how the pore pressure induced by the fluid can reduce basal friction, and increase the run-out length of the avalanche. Bouchut, Fernández-Nieto, et al. (2016) construct a two-phase two-layers shallow model taking into account dilatancy effects with the model from Roux and Radjai (1998).

The granular collapse simulations by Lagrée et al. (2011) were also done using a multiphase solver, Gerris[1]; however, the granular phase was considered as fully incompressible, while here we will attempt to model unilateral incompressibility.

Computer graphics Müller et al. (2005) proposed a first approach for simulating interacting fluids within the SPH framework, but considered a single velocity field for the different phases. H. Zhu et al. (2006) presented simulations of general mixtures using a Lattice-Boltzmann approach. L. Boyd and Bridson (2012) introduced a FLIP extension for multiphase immiscible flows able two take into account a distinct velocity for each phase. Nielsen and Østerby (2013) simulated the coupled motion of droplets of water in air, with a convection-diffusion process for the water phase. Ren et al. (2014) developed an extension of SPH targeted at mixtures.

More relevant to our applications, Lenaerts and Dutré (2009) demonstrated a first approach for the simulation of the interactions between water and a granular continuum. To do so, they coupled a granular SPH model with the framework of (Lenaerts, Adams, et al. 2008), which simulates the interaction of a fluid with porous materials by adaptively absorbing and recreating fluid particles, with a diffusive model for the transport of the fluid inside the porous material. Their approach was able to capture the transition of a granular material into mud, but cannot reproduce sedimentation processes.

Discrete simulations Finally, while our goal is to simulate systems with a large enough number of grains to motivate the continuum approach, we can mention that several works have focused on coupled fluid–granular simulations using diverse discrete models for the particles.

Choi and Joseph (2001) performed direct numerical simulations of the fluidization of a few hundred of particles, adapting the simulation mesh to their boundaries. Pignatel et al. (2009) simulated the low-velocity, gravity-driven injection of particles inside a Newtonian fluid using particle sampling and compared their results to both experiments and a continuum model using the effective viscosity from Krieger (1972) to model the stress in the particulate phase. Topin, Dubois, et al. (2011) and Topin, Monerie, et al. (2012) couple the Non-Smooth Contact Dynamics method for the granular phase with a Newtonian fluid simulation using the fictitious domain approach (that is, extending the fluid domain to the interior of the particles and enforcing rigid motion there). Maurin et al. (2015) used an hybrid approach, using mixture theory for the surrounding fluid yet computing the interactions of the particles with a molecular dynamics approach similar to that of (Cundall and Strack 1979).

Our approach Despite the variety of methods mentioned above, none of them proposes a fully coupled continua simulation while simultaneously enforcing the non-associated Drucker–Prager

[1] http://gfs.sourceforge.net

flow rule for the granular phase, without regularization of the yield stress. Yet, such a nonsmooth treatment is necessary to avoid the lingering of residual velocities (at the end of an avalanche, for instance). To achieve this goal, we will extend the framework of Chapter 7 to account for the Newtonian fluid. More precisely, we will make use of the two-phase model of Anderson and Jackson (1967) with the drag expression of Richardson and Zaki (1954). To construct our system of dimensionless equations, we will also take inspiration from Boyer (2001). Indeed, while this latter work deals with the simulation of immiscible fluids with surface tension, it uses a two-phase approach that is relevant for us.

8.2 Two-phase model

In this section, we shall propose equations for the motion of a granular material subject to a Drucker–Prager or $\mu(I)$ yield stress inside a Newtonian fluid, following the approach of Anderson and Jackson (1967) and Jackson (2000). Our overview of the modeling literature has shown that the expression of the stresses acting on the two phases is not fully known, and that several alternatives have been proposed even for the study of similar phenomena. Here we will attempt to make reasonable choices, in the hope that they will prove relevant enough to capture the qualitative behavior of the materials, but are fully aware that comparisons with experiments remain necessary in order to determine definitive expressions for those stresses.

8.2.1 Base equations

We start from the momentum and mass conservation equations for the granular and fluid continua. We assume that the sole external force applied our diphasic medium is gravity.

$$\rho_g \phi \frac{D_{\mathbf{u}_g} \mathbf{u}_g}{D_{\mathbf{u}_g} t} - \nabla \cdot \left[\boldsymbol{\sigma}_g \right] = \rho_g \phi \boldsymbol{g} + \boldsymbol{f}_{f \to g} \tag{8.4}$$

$$\rho_f (1-\phi) \frac{D_{\mathbf{u}_f} \mathbf{u}_f}{D_{\mathbf{u}_f} t} - \nabla \cdot \left[\boldsymbol{\sigma}_f \right] = \rho_f (1-\phi) \boldsymbol{g} - \boldsymbol{f}_{f \to g} \tag{8.5}$$

$$\frac{\partial \phi}{\partial t} + \nabla \cdot \left[\phi \mathbf{u}_g \right] = 0 \tag{8.6}$$

$$\frac{\partial (1-\phi)}{\partial t} + \nabla \cdot \left[(1-\phi) \mathbf{u}_f \right] = 0 \tag{8.7}$$

Expressions for the stresses $\boldsymbol{\sigma}_{f,g}$ and interfacial momentum transfer $\boldsymbol{f}_{f \to g}$ term remain to be written. As we have seen in our brief literature review, these have been subject to thorough debate in the last decades. Moreover, different physical considerations may lead to attribute certain contributions to one term or the other, but still lead to the same equations. Here we shall follow Anderson and Jackson (1967), and state that the interfacial momentum transfer should consist of a drag term $\boldsymbol{f}^d_{f \to g}$ and a generalized buoyancy contribution $\boldsymbol{f}^b_{f \to g}$,

$$\boldsymbol{f}_{f \to g} := \boldsymbol{f}^d_{f \to g} + \boldsymbol{f}^b_{f \to g}.$$

Stresses will include a pressure term, a contribution from the Newtonian mixture viscosity, and the stress induced by the contact between particles.

8.2.2 Stresses and buoyancy

We assume the following expression for the stresses:

$$\begin{aligned} \boldsymbol{\sigma}_g &= -p_g \mathbb{I} + \boldsymbol{\sigma}^v_g + \phi \boldsymbol{\sigma}^C \\ \boldsymbol{\sigma}_f &= -p_f \mathbb{I} + \boldsymbol{\sigma}^v_f. \end{aligned}$$

$p_{f,g}$ corresponds to the partial pressure in each phase, $\boldsymbol{\sigma}^{v}_{f,g}$ to Newtonian viscosities, and $\boldsymbol{\sigma}^{C}$ to the supplemental stress due to the contacts between grains (the effective stress, in Terzaghi's terms).

Our first assumption, which follows (Chauchat, Guillou, et al. 2013; Einstein 1906; Jackson 2000), will be that the Newtonian viscous forces should be relative to the volume-average velocity $\mathbf{u}_{\mathrm{v}}$. Moreover, the viscosity shall depend only on the local volume fraction, i.e., $\boldsymbol{\sigma}_{f,g} = \eta_{f,g}(\phi)\,\mathrm{D}(\mathbf{u}_{\mathrm{v}})$.

Pore stress We first consider the stress conveyed at the interface between the two phases; this is the case for the pressure $p_{f,g}$ and viscous stresses $\boldsymbol{\sigma}^{v}_{f,g}$. Following Anderson and Jackson (1967), these stresses generate a generalized buoyancy force density,

$$\boldsymbol{f}^{b}_{f\to g} = -\phi\nabla\cdot\left[\boldsymbol{\sigma}^{b}_{f}\right] + (1-\phi)\nabla\cdot\left[\boldsymbol{\sigma}^{b}_{g}\right],$$

where $\boldsymbol{\sigma}^{b}_{f,g} := -p_{f,g}\mathbb{I} + \boldsymbol{\sigma}^{v}_{f,g}$. Note that this is consistent with, and a generalization of, the classical Archimede's principle.

Now, as mentioned by Bürger (2000), $\boldsymbol{\sigma}^{b}_{f,g}$ are mathematical quantities that cannot be experimentally measured, in contrast to the *pore* (or interstitial) stress $\boldsymbol{\sigma}^{b}$, which can. Indeed, following from Terzaghi's principle, $\boldsymbol{\sigma}^{b}$ can be computed as the difference of the total stress and the effective stress due to contacts, i.e., $\boldsymbol{\sigma}^{b} = (\boldsymbol{\sigma}_{g} + \boldsymbol{\sigma}_{f}) - \phi\boldsymbol{\sigma}^{C}$. When the material is at rest, $\boldsymbol{\sigma}^{b}$ simply corresponds to the hydrostatic pressure. We can make two kinds of hypothesis to relate $\boldsymbol{\sigma}^{b}_{f,g}$ to $\boldsymbol{\sigma}^{b}$, which, when taking into account buoyancy, will ultimately lead to the same set of equations.

- We can suppose that the viscosity and pressure belong fully to the fluid phase, as done by e.g., Pitman and Le (2005). This means $\boldsymbol{\sigma}^{b}_{g} = \mathbf{0}$, and $\boldsymbol{\sigma}^{b}_{f} = \boldsymbol{\sigma}^{b}$. In this case, the generalized buoyancy is simply $\boldsymbol{f}^{b}_{f\to g} = -\phi\nabla\cdot\left[\boldsymbol{\sigma}^{b}\right]$.
- Alternatively, we can suppose that the phase stresses $\boldsymbol{\sigma}^{b}_{f,g}$ are given by pore stress $\boldsymbol{\sigma}^{b}_{f,g}$ scaled with their relative volume fractions, that is $\boldsymbol{\sigma}^{b}_{g} = \phi\boldsymbol{\sigma}^{b}$ and $\boldsymbol{\sigma}^{b}_{f} = (1-\phi)\boldsymbol{\sigma}^{b}$. Then the generalized buoyancy is $\boldsymbol{f}^{b}_{f\to g} = -\phi\nabla\cdot\left[(1-\phi)\boldsymbol{\sigma}^{b}\right] + (1-\phi)\nabla\cdot\left[\phi\boldsymbol{\sigma}^{b}\right] = \nabla\cdot\left[\phi\boldsymbol{\sigma}^{b}\right] - \phi\nabla\cdot\left[\boldsymbol{\sigma}^{b}\right] = \boldsymbol{\sigma}^{v}\nabla\phi$.

In both cases, the pore stress can be decomposed as $\boldsymbol{\sigma}^{b} = \mathrm{p}\mathbb{I} - \eta_{\mathrm{eff}}(\phi)\,\mathrm{D}(\mathbf{u}_{\mathrm{v}})$, where p is called the pore pressure, and we get:

$$\begin{aligned}
\nabla\cdot\left[\boldsymbol{\sigma}^{b}_{g}\right] - \boldsymbol{f}_{f\to g} &= \phi\boldsymbol{\sigma}^{b} = \phi\left(\eta_{\mathrm{eff}}(\phi)\,\mathrm{D}(\mathbf{u}_{\mathrm{v}}) - \mathrm{p}\mathbb{I}\right)\\
\nabla\cdot\left[\boldsymbol{\sigma}^{b}_{f}\right] + \boldsymbol{f}_{f\to g} &= (1-\phi)\boldsymbol{\sigma}^{b} = (1-\phi)\left(\eta_{\mathrm{eff}}(\phi)\,\mathrm{D}(\mathbf{u}_{\mathrm{v}}) - \mathrm{p}\mathbb{I}\right).
\end{aligned}$$

Equations (8.4–8.5) can thus be rewritten as

$$\rho_{g}\phi\frac{D_{\mathbf{u}_{g}}\mathbf{u}_{g}}{D_{\mathbf{u}_{g}}t} - \phi\nabla\cdot\left[\eta_{\mathrm{eff}}(\phi)\,\mathrm{D}(\mathbf{u}_{\mathrm{v}}) - \mathrm{p}\mathbb{I}\right] - \nabla\cdot\left[\phi\boldsymbol{\sigma}^{C}\right] = \rho_{g}\phi\boldsymbol{g} + \boldsymbol{f}^{d}_{f\to g} \tag{8.8}$$

$$\rho_{f}(1-\phi)\frac{D_{\mathbf{u}_{f}}\mathbf{u}_{f}}{D_{\mathbf{u}_{f}}t} - (1-\phi)\nabla\cdot\left[\eta_{\mathrm{eff}}(\phi)\,\mathrm{D}(\mathbf{u}_{\mathrm{v}}) - \mathrm{p}\mathbb{I}\right] = \rho_{f}(1-\phi)\boldsymbol{g} - \boldsymbol{f}^{d}_{f\to g} \tag{8.9}$$

It remains to choose an expression for $\eta_{\mathrm{eff}}(\phi)$. We can for instance suppose η_{eff} to be constant, or use Einstein's expression, $\eta_{\mathrm{eff}}(\phi) = \eta_{f}\left(1 + \frac{5}{2}\phi\right)$.

Contact stress Our second main assumption is that the contacts between grains are subject to dry friction. Just like in the dry case of Chapter 7, the grains will be considered to be self-contacting, with no layer of fluid between them, as soon as the maximal volume fraction is reached.

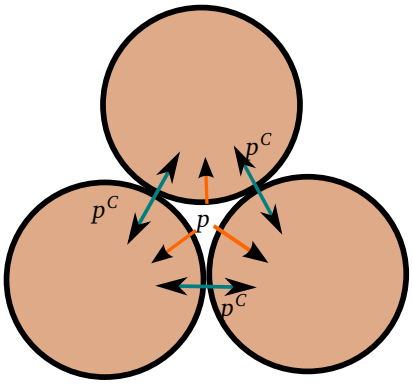


Figure 8.1: *Two kinds of competing normal stresses acting on the particles: the pore pressure* p *and the effective stress due to contacts* $p^C = -\frac{1}{d}\operatorname{Tr}\boldsymbol{\sigma}^c$

As in the dry case, we will want the grains volume fraction ϕ, contact stress $\boldsymbol{\sigma}^C$, and strain rate $\dot{\boldsymbol{\varepsilon}}_g$, to follow the maximal volume fraction constraint with non-associated Drucker–Prager flow rule, i.e.,

$$\begin{cases} 0 \le p^C \perp \phi_{\max} - \phi \ge 0 & \\ \operatorname{Dev}\boldsymbol{\sigma}^C = \hat{\mu} p^C \dfrac{\operatorname{Dev}\dot{\boldsymbol{\varepsilon}}_g}{|\operatorname{Dev}\dot{\boldsymbol{\varepsilon}}_g|} & \text{if } \operatorname{Dev}\dot{\boldsymbol{\varepsilon}}_g \neq \mathbf{0} \text{ (yielded)} \\ |\operatorname{Dev}\boldsymbol{\sigma}^C| \le \hat{\mu} p^C & \text{if } \operatorname{Dev}\dot{\boldsymbol{\varepsilon}}_g = \mathbf{0} \text{ (unyielded),} \end{cases} \tag{8.10}$$

where $p^C := -\frac{1}{d}\operatorname{Tr}\boldsymbol{\sigma}^C$.

This solid-contacts assumption also means that the stress due to contacts between grains will be propagated at grain–grain interfaces only, and exist solely inside the granular phase. As such, there is no associated buoyancy force. This is confirmed by a simple thought experiment: consider a dry, compact heap of rigid grains at rest, supporting a given mass m. Modifying the intensity of the contact forces between the grains, for instance by changing m, will not affect the air phase nor the interstitial pressure p.

Remark 8.1. *We can already remark that as illustrated on Figure 8.1, the normal component of the total stress of the mixture will be the sum of two terms,* p *and* $p^C = -\frac{1}{d}\operatorname{Tr}\boldsymbol{\sigma}^C$*. Now, as per the Drucker–Prager criterion (8.10), increasing* p^C *increases the yield stress of the material; for a given applied load, the pore pressure* p *will thus have a weakening effect.*

8.2.3 Drag force

Our third core hypothesis will be to consider grains that are thin enough, or with velocity (w.r.t. the surrounding fluid) small enough that Stokes' drag law apply. That is, we consider the case where the grain-level Reynolds number, $\mathrm{Re}_g := \frac{\rho_f W D_g}{\eta_f}$, where D_g is the diameter of the grains and W a characteristic relative velocity, is small. This choice has been made by numerous authors, e.g., (Anderson and Jackson 1967; Bürger 2000; Chauchat and Médale 2010; Harris and Crighton 1994; Pitman and Le 2005) , while others consider the drag force to be quadratic in the relative velocity (Batchelor 1988; Chauchat, Guillou, et al. 2013; Drew 1983). We will stick with simple linear drag, knowing that in any case we can account for the nonlinearity by explicitly adjusting the drag coefficient in the numerical method, and remember that we may overestimate the relative velocity in the analysis below.

We will thus express the drag force as $\boldsymbol{f}_{f\to g} = -\hat{\xi}(\phi)\mathbf{w}$, where $\mathbf{w}$ is the relative velocity between the two phases, $\mathbf{w} := \mathbf{u}_g - \mathbf{u}_f$, and $\hat{\xi}$ is a scalar function of the volume fraction. In the

two-phase literature, $\hat{\xi}(\phi)$ is commonly deduced from sedimentation experiments, for instance using the empirical law of Richardson and Zaki (1954), such as in (Chauchat, Guillou, et al. 2013). The quasistatic conservation of momentum for the sedimenting particles is written as

$$0 = \phi(\rho_g - \rho_f)g - \hat{\xi}(\phi)u_z(\phi),$$

yielding

$$\hat{\xi}(\phi) = \frac{\phi(\rho_g - \rho_f)g}{u_z(\phi)}.$$

Equation (8.1) gives $u_z(\phi) = w_\infty(1-\phi)^\nu$; let us compute the value of w_∞ under the hypothesis $\mathrm{Re}_g \leq 1$. Let w denote the relative velocity of a single spherical grain falling through the fluid. Stokes' law state that the intensity of the drag on the particle is $-3\pi\eta_f D_g w$, meaning that the equilibrium is reached for $(\rho_g - \rho_f)(\frac{4}{3}\pi\left(\frac{D_g}{2}\right)^3)g = 3\pi\eta_f D_g w$, i.e.,

$$w_\infty = \frac{(\rho_g - \rho_f)gD_g^2}{18\eta_f}.$$

Overall, we conclude

$$\hat{\xi}(\phi) = \eta_f \frac{18}{D_g^2}\phi(1-\phi)^{-\nu}.$$

Note that this expression increases monotonically with the volume fraction of grains, which at first may appear strange, as one expects the average drag force to decrease for denser objects. However, this weakening of the drag force is due to a smaller relative velocity, as the fluid is lugged by the higher density of grains, rather than a decrease in the friction coefficient. Note also that this expression makes $\hat{\xi}$ grow to infinity when the fluid vanishes, thus requiring the relative velocity $\mathbf{w}$ to vanish for purely granular materials. As our maximal volume fraction constraint requires $\phi \leq \phi_{\max} < 1$, we need not worry about this degenerate case, however. Other closures are possible; for instance, Chauchat and Médale (2010) use the so-called *Carman–Kozeny* relationship. For now, we will not assume a precise expression for $\hat{\xi}$, but simply write that

$$\boldsymbol{f}^d_{f\to g} = -\phi(1-\phi)\xi(\phi)\mathbf{w}, \tag{8.11}$$

and remember that $\xi(\phi)$ in Pa.s.m^{-2} is similar in order of magnitude to $g\frac{\rho_g-\rho_f}{w_\infty(1-\phi)^{\nu+1}} = \frac{\eta_f}{D_g^2}\frac{18}{(1-\phi)^{\nu+1}}$.

Summary of our hypothesis At this point, we have made the following hypothesis:

- no mass is transferred between the two phases;
- the Newtonian viscous stress is proportional to $\mathrm{D}(\mathbf{u}_\mathrm{v})$;
- the linear Stokes drag law applies;
- the contacts between grains are subject to dry friction and occur when $\phi = \phi_{\max} < 1$.

8.2.4 Mixture conservation equations

The conservation equations (8.4–8.7) written with the phase velocities are not very convenient; Boyer (2001) propose to express them instead as functions of two other velocity variables, the volume-average velocity $\mathbf{u}_\mathrm{v}$ and the relative velocity $\mathbf{w}$. This choice does have some nice properties; the Newtonian viscosity is already expressed as a function of $\mathbf{u}_\mathrm{v}$, and $\mathbf{u}_\mathrm{v}$ is the only velocity variable that is divergence-free. Indeed, summing Equations (8.6) and (8.7), we get

$$\nabla\cdot\mathbf{u}_\mathrm{v} = -\frac{\partial 1}{\partial t} = 0. \tag{8.12}$$

However, using $\mathbf{u}_\mathrm{v}$ also requires Boyer (2001) to deal with rather inconvenient inertial terms. Here, we will choose instead to use $\mathbf{u}_\mathrm{m}$ and $\mathbf{w}$ as our velocity variables.

Notations Following Etienne (2004), we introduce the scaled density difference

$$\alpha := \frac{\rho_g - \rho_f}{\rho_f}.$$

Note that $\rho_g/\rho_f = (\alpha+1)$. Let $\beta(\phi) := (1+\alpha\phi)$, so that the total density of the mixture is given by $\rho(\phi) = \phi\rho_g + (1-\phi)\rho_f = \beta(\phi)\rho_f$. Let $\pi(\phi) := \phi(1-\phi)$.

The total mass conservation of the mixture, obtained by summing (8.6) scaled by ρ_g and (8.7) scaled by ρ_f, reads

$$\frac{\partial \beta}{\partial t} + \nabla\cdot[\beta \mathbf{u}_\mathrm{m}] = 0. \tag{8.13}$$

Finally, note that phase and volume-averaged velocities can also be expressed as function of the mass-averaged and relative velocities:

$$\mathbf{u}_g = \mathbf{u}_\mathrm{m} + \frac{(1-\phi)}{\beta}\mathbf{w} \qquad \mathbf{u}_f = \mathbf{u}_\mathrm{m} - (\alpha+1)\frac{\phi}{\beta}\mathbf{w}$$
$$\mathbf{u}_\mathrm{v} = \mathbf{u}_\mathrm{m} - \alpha\frac{\pi}{\beta}\mathbf{w}$$

Mixture momentum conservation To obtain the conservation equation for the total momentum of the mixture, we just have to sum Equations (8.8) and (8.9). It is simpler to consider the conservative expression[2] for the inertial terms,

$$\phi\frac{D_{\mathbf{u}_g}\mathbf{u}_g}{D_{\mathbf{u}_g}t} = \frac{\partial \phi\mathbf{u}_g}{\partial t} + \nabla\cdot\left[\phi\mathbf{u}_g\otimes\mathbf{u}_g\right]$$
$$(1-\phi)\frac{D_{\mathbf{u}_f}\mathbf{u}_f}{D_{\mathbf{u}_f}t} = \frac{\partial (1-\phi)\mathbf{u}_g}{\partial t} + \nabla\cdot\left[(1-\phi)\mathbf{u}_f\otimes\mathbf{u}_f\right].$$

Remark first that

$$\begin{aligned}
&\rho_g\phi\mathbf{u}_g\otimes\mathbf{u}_g + \rho_f(1-\phi)\mathbf{u}_f\otimes\mathbf{u}_f \\
&\quad= \rho_f\left[(\alpha+1)\phi\mathbf{u}_g\otimes\mathbf{u}_\mathrm{m} + (1-\phi)\mathbf{u}_f\otimes\mathbf{u}_\mathrm{m}\right] \\
&\qquad+ \rho_f\left[(\alpha+1)\phi\mathbf{u}_g\otimes(\mathbf{u}_g-\mathbf{u}_\mathrm{m}) + (1-\phi)\mathbf{u}_f\otimes(\mathbf{u}_f-\mathbf{u}_\mathrm{m})\right] \\
&\quad= \rho_f\left[\beta\mathbf{u}_\mathrm{m}\otimes\mathbf{u}_\mathrm{m} + (\alpha+1)\frac{\phi(1-\phi)}{\beta}\mathbf{u}_g\otimes\mathbf{w} - (\alpha+1)\frac{\phi(1-\phi)}{\beta}\mathbf{u}_f\otimes\mathbf{w}\right] \\
&\quad= \rho_f\left[\beta\mathbf{u}_\mathrm{m}\otimes\mathbf{u}_\mathrm{m} + (\alpha+1)\frac{\pi}{\beta}\mathbf{w}\otimes\mathbf{w}\right].
\end{aligned}$$

Therefore,

$$\begin{aligned}
&\rho_g\phi\frac{D_{\mathbf{u}_g}\mathbf{u}_g}{D_{\mathbf{u}_g}t} + \rho_f(1-\phi)\frac{D_{\mathbf{u}_f}\mathbf{u}_f}{D_{\mathbf{u}_f}t} \\
&\quad= \rho_f\left[\frac{\partial\beta\mathbf{u}_\mathrm{m}}{\partial t} + (\beta\mathbf{u}_\mathrm{m}\cdot\nabla)\mathbf{u}_\mathrm{m} + \left\langle \underbrace{\nabla\cdot[\beta\mathbf{u}_\mathrm{m}]}_{-\frac{\partial\beta}{\partial t}}, \mathbf{u}_\mathrm{m}\right\rangle + (\alpha+1)\nabla\cdot\left[\frac{\pi}{\beta}\mathbf{w}\otimes\mathbf{w}\right]\right] \\
&\quad= \rho_f\left[\beta\frac{D_{\mathbf{u}_\mathrm{m}}\mathbf{u}_\mathrm{m}}{D_{\mathbf{u}_\mathrm{m}}t} + (\alpha+1)\nabla\cdot\left[\frac{\pi}{\beta}\mathbf{w}\otimes\mathbf{w}\right]\right].
\end{aligned}$$

The conservation equation for the total momentum of the mixture thus reads

$$\rho_f\left[\beta\frac{D_{\mathbf{u}_\mathrm{m}}\mathbf{u}_\mathrm{m}}{D_{\mathbf{u}_\mathrm{m}}t} + (\alpha+1)\nabla\cdot\left[\frac{\pi}{\beta}\mathbf{w}\otimes\mathbf{w}\right]\right] + \nabla\mathrm{p} - \nabla\cdot\left[\eta_\mathrm{eff}\,\mathrm{D}(\mathbf{u}_\mathrm{m} - \alpha\frac{\pi}{\beta}\mathbf{w})\right] - \nabla\cdot\left[\phi\boldsymbol{\sigma}^C\right] = \rho_f\beta\boldsymbol{g} \tag{8.14}$$

[2] see Section 5.2.1

Fluctuation momentum conservation Another interesting linear combination of the momentum conservation consists in taking the difference of (8.8) scaled by $\rho_f(1-\phi)$, and (8.9) scaled by $\phi\rho_g$. Processing term by term, we have

- Inertial terms:

$$\rho_g\rho_f\pi\frac{D_{\mathbf{u}_g}\mathbf{u}_g}{D_{\mathbf{u}_g}t}-\rho_g\rho_f\pi\frac{D_{\mathbf{u}_f}\mathbf{u}_f}{D_{\mathbf{u}_f}t}=\rho_g\rho_f\pi\left[\frac{\partial\mathbf{w}}{\partial t}+(\mathbf{u}_g\cdot\nabla)\mathbf{u}_g-(\mathbf{u}_f\cdot\nabla)\mathbf{u}_f\right]$$
$$=\rho_g\rho_f\pi\left[\frac{\partial\mathbf{w}}{\partial t}+\left(\langle\!|\mathbf{u}_{f,g}|\!\rangle\cdot\nabla\right)\mathbf{w}-(\mathbf{w}\cdot\nabla)\langle\!|\mathbf{u}_{f,g}|\!\rangle\right],$$
$$\text{with } \langle\!|\mathbf{u}_{f,g}|\!\rangle:=\frac{\mathbf{u}_f+\mathbf{u}_g}{2}=\mathbf{u}_\mathrm{m}+\frac{1-2\phi-\alpha\phi}{2\beta}\mathbf{w};$$

- Mixture stress:

$$-\rho_f\pi\nabla\cdot\left[\boldsymbol{\sigma}^b\right]+\rho_g\pi\nabla\cdot\left[\boldsymbol{\sigma}^b\right]=\rho_f\alpha\pi\nabla\cdot\left[\boldsymbol{\sigma}^b\right];$$

- Drag force:

$$\rho_f(1-\phi)\boldsymbol{f}^d_{f\to g}+\rho_g\phi\boldsymbol{f}^d_{f\to g}=\rho_f\beta\boldsymbol{f}^d_{f\to g}=-\rho_f\beta\pi\xi\mathbf{w};$$

- Gravity:

$$\rho_f(1-\phi)\phi\rho_g\boldsymbol{g}-\rho_g\phi\rho_f(1-\phi)\boldsymbol{g}=\mathbf{0}.$$

The conservation of the fluctuation momentum, where we have divided each term by $(1-\phi)\rho_f$[3], thus reads

$$(\alpha+1)\rho_f\phi\left[\frac{\partial\mathbf{w}}{\partial t}+\left(\langle\!|\mathbf{u}_{f,g}|\!\rangle\cdot\nabla\right)\mathbf{w}+(\mathbf{w}\cdot\nabla)\langle\!|\mathbf{u}_{f,g}|\!\rangle\right]+\beta\phi\xi\mathbf{w}$$
$$-\alpha\phi\nabla\mathrm{p}+\alpha\phi\nabla\cdot\left[\eta_\mathrm{eff}\mathrm{D}(\mathbf{u}_\mathrm{m}-\alpha\frac{\pi}{\beta}\mathbf{w})\right]-\nabla\cdot\left[\phi\boldsymbol{\sigma}^C\right]=\mathbf{0}. \quad (8.15)$$

Remark The velocity transport terms in Equations (8.14) and (8.15) seem relatively unwieldy, and, as we will see in the next section, may not be negligible for certain applications. However, their expression will be drastically simplified when using a timestepping algorithm with an explicit velocity-advection scheme, such as the characteristics method or particle-based transport.

8.2.5 Dimensionless equations

Characteristic quantities To get an idea of the respective importance of the different terms in Equation (8.14)and (8.15), we need to define characteristic values for our variables. There are several ways to do this, and none will hold for the whole range of regimes that can be exhibited by our two-phase material. As we will mostly be studying gravity-driven flows, we define the characteristic mixture velocity as $U:=\sqrt{gL}$, where L is a characteristic length of the studied phenomenon. For the same reason, we choose $P:=\rho_f gL$ as the characteristic pore pressure, $(\alpha+1)P=\rho_g gL$ as the characteristic contact stress, $T=\frac{L}{U}$ as the characteristic time. Finally, at the risk of overestimating W, we define the characteristic relative velocity as the sedimentation velocity of a single grain, w_∞.

The dimensionless variables will be denoted with a tilde in the equations below, but we will quickly drop their decoration to lighten notation. We also introduce dimensionless versions of the effective viscosity and drag fields,

$$\eta_\mathrm{eff}=\eta_f\tilde{\eta_\mathrm{eff}} \qquad\qquad \xi=g\frac{\rho_g-\rho_f}{w_\infty}\Xi\tilde{\eta_\mathrm{eff}},$$

where the dimensionless number Ξ denote the order of magnitude of the geometry dependent term in ξ, i.e., $\Xi\sim\xi\frac{w_\infty}{\rho_g-\rho_f}g\sim(1-\phi)^{-\nu-1}$ when using the Richardson and Zaki (1954) closure. Note that as ν is usually taken to be greater than 3, the choice of Ξ will be highly dependent of the target volume fraction.

[3] We supposed that $\phi_\mathrm{max}<1$ and thus $\phi=1$ is never reached

Mixture momentum conservation The dimensionless version of (8.14) is, after division by ρ_f,

$$\frac{1}{L}\left[\beta U^2\frac{D\tilde{\mathbf{u}_m}}{D\tilde{t}}+(\alpha+1)UW\tilde{\nabla}\cdot\left[\frac{\pi}{\beta}\tilde{\mathbf{w}}\otimes\tilde{\mathbf{w}}\right]\right]-(\alpha+1)g\tilde{\nabla}\cdot\left[\phi\tilde{\boldsymbol{\sigma}}^C\right]$$
$$+g\tilde{\nabla}\mathrm{p}-\frac{\eta_f}{\rho_f L^2}\tilde{\nabla}\cdot\left[\tilde{\eta_{\mathrm{eff}}}\tilde{\mathrm{D}}(U\tilde{\mathbf{u}_m}-W\alpha\frac{\pi}{\beta}\tilde{\mathbf{w}})\right]=g\beta\boldsymbol{e}_g.$$

This motivates the introduction of two dimensionless number; the Reynolds number of the fluid, Re, and a Stokes number, St, relating the importance of the mixture kinetic energy to the dissipation by drag forces:

$$\mathrm{Re}:=\frac{\rho_f UL}{\eta_f}\qquad\qquad \mathrm{St}:=\frac{W}{U}=\frac{(\rho_g-\rho_f)gD_g^2}{18U\eta_f}=\frac{\rho_g UD_g^2}{18L\eta_f}.$$

The relationship between these dimensionless numbers is given by ratios of density and length,

$$\mathrm{St}=\frac{\alpha+1}{18}\frac{\rho_f UD_g^2}{L\eta_f}=\frac{\alpha+1}{18}\mathrm{Re}\left(\frac{D_g}{L}\right)^2=\frac{\alpha+1}{18}\epsilon^2\mathrm{Re},$$

where $\epsilon:=\frac{D_g}{L}$ denote the ratio of the grains diameter to the characteristic simulation length.

Dividing our equation by $g(\alpha+1)$, we get the dimensionless momentum conservation equation for the mixture,

$$\frac{\beta}{\alpha+1}\frac{D\tilde{\mathbf{u}_m}}{D\tilde{t}}+\mathrm{St}^2\tilde{\nabla}\cdot\left[\frac{\pi}{\beta}\tilde{\mathbf{w}}\otimes\tilde{\mathbf{w}}\right]-\tilde{\nabla}\cdot\left[\phi\tilde{\boldsymbol{\sigma}}^C\right]$$
$$-\frac{1}{\alpha+1}\tilde{\nabla}\cdot\left[\frac{\tilde{\eta_{\mathrm{eff}}}}{\mathrm{Re}}\tilde{\mathrm{D}}(\tilde{\mathbf{u}_m}-\mathrm{St}\alpha\frac{\pi}{\beta}\tilde{\mathbf{w}})-\tilde{\mathrm{p}}\mathbb{I}\right]=\frac{\beta}{\alpha+1}\boldsymbol{e}_g. \quad (8.16)$$

Fluctuation momentum conservation Rewriting (8.15) using our dimensionless variables yield

$$(\alpha+1)\rho_f\phi\frac{W}{L}\left[U\frac{\partial\tilde{\mathbf{w}}}{\partial\tilde{t}}+UI(\tilde{\mathbf{u}_m},\tilde{\mathbf{w}})+WI(\frac{1-2\phi-\alpha\phi}{2\beta}\tilde{\mathbf{w}},\tilde{\mathbf{w}})\right]+\beta\phi g\frac{\rho_g-\rho_f}{w_\infty}\Xi W\tilde{\xi}\tilde{\mathbf{w}}$$
$$=\alpha\phi\left(g\rho_f\nabla\tilde{\mathrm{p}}-\frac{\eta_f}{L^2}\nabla\cdot\left[\tilde{\eta_{\mathrm{eff}}}\mathrm{D}(U\tilde{\mathbf{u}_m}-\alpha W\frac{\pi}{\beta}\tilde{\mathbf{w}})\right]\right)+(\alpha+1)\rho_f g\tilde{\nabla}\cdot\left[\phi\tilde{\boldsymbol{\sigma}}^C\right],$$

where $I(\mathbf{v},\mathbf{w}):=(\mathbf{v}\cdot\tilde{\nabla})\mathbf{w}+(\mathbf{w}\cdot\tilde{\nabla})\mathbf{v}$. Dividing both sides by $\rho_g g$, and remarking that $WU/L=\mathrm{St}g$, we get

$$\phi\mathrm{St}\left[\frac{\partial\tilde{\mathbf{w}}}{\partial\tilde{t}}+I(\tilde{\mathbf{u}_m},\tilde{\mathbf{w}})+\mathrm{St}I(\frac{1-2\phi-\alpha\phi}{2\beta}\tilde{\mathbf{w}},\tilde{\mathbf{w}})\right]+\frac{\alpha\beta}{\alpha+1}\phi\Xi\tilde{\xi}\tilde{\mathbf{w}}$$
$$=\frac{\alpha\phi}{\alpha+1}\left(\tilde{\nabla}\tilde{\mathrm{p}}-\frac{1}{\mathrm{Re}}\tilde{\nabla}\cdot\left[\tilde{\eta_{\mathrm{eff}}}\tilde{\mathrm{D}}(\tilde{\mathbf{u}_m}-\alpha\mathrm{St}\frac{\pi}{\beta}\tilde{\mathbf{w}})\right]\right)+\tilde{\nabla}\cdot\left[\phi\tilde{\boldsymbol{\sigma}}^C\right]. \quad (8.17)$$

Mass conservation The dimensionless mass conservation equations read:

$$\tilde{\nabla}\cdot\left[\tilde{\mathbf{u}_m}-\mathrm{St}\frac{\alpha\pi}{\beta}\tilde{\mathbf{w}}\right]=0 \quad (8.18)$$

$$\frac{\partial\beta}{\partial\tilde{t}}-\tilde{\nabla}\cdot[\beta\tilde{\mathbf{u}_m}]=0. \quad (8.19)$$

Alternatively, Equation (8.19) can be replaced with (8.20),

$$\frac{\partial\phi}{\partial\tilde{t}}-\tilde{\nabla}\cdot\left[\phi\tilde{\mathbf{u}}_g\right]=0, \quad (8.20)$$

where $\tilde{\mathbf{u}}_g:=\frac{1}{U}\mathbf{u}_g$.

8.2.6 Particular cases

Equilibrium First, let us look at static equilibrium, which means $\mathbf{u}_{\mathrm{m}} = \mathbf{w} = \mathbf{0}$. The dimensionless conservation equations read simply

$$\begin{cases} -\tilde{\nabla}\cdot\left[\phi\tilde{\boldsymbol{\sigma}}^{C}\right] + \dfrac{1}{\alpha+1}\tilde{\nabla}\tilde{\mathrm{p}} = \dfrac{\beta}{\alpha+1}\boldsymbol{e}_g \\ -\tilde{\nabla}\cdot\left[\phi\tilde{\boldsymbol{\sigma}}^{C}\right] = \dfrac{\alpha\phi}{\alpha+1}\tilde{\nabla}\tilde{\mathrm{p}}, \end{cases}$$

from which we deduce easily

$$\begin{cases} \tilde{\nabla}\tilde{\mathrm{p}} = \boldsymbol{e}_g \\ -\tilde{\nabla}\cdot\left[\phi\,\tilde{\boldsymbol{\sigma}}^{C}\right] = \dfrac{\alpha\phi}{\alpha+1}\boldsymbol{e}_g. \end{cases}$$

Hence, up to a constant field, the pore pressure satisfies $\mathrm{p} = \rho_f g z$, and corresponds to the expected hydrostatic pressure. We also have $-\nabla\cdot\left[\phi\boldsymbol{\sigma}_g\right] = \phi\alpha\rho_f\boldsymbol{g} = \phi\rho_g\boldsymbol{g} - \phi\nabla\mathrm{p}$; the contact force thus oppose the action of gravity on the grains reduced by the buoyancy force given by Archimede's principle. Again, this is the expected result.

Incompressible flow The divergence-free condition for the volume-averaged velocity $\mathbf{u}_{\mathrm{v}}$ is expressed by Equation (8.18).

In the limit case where $\mathrm{St} = 0$, the mass-averaged velocity $\mathbf{u}_{\mathrm{m}}$ becomes also divergence-free. In a similar fashion, if $\Xi = +\infty$, then (8.17) imposes $\mathbf{w} = \mathbf{0}$ and then once again the flow becomes incompressible.

Single phase limit If we consider that the surrounding fluid is massless and inviscid, $\alpha = +\infty$, and $\frac{\beta}{\alpha+1} = \phi$. Moreover, if the grains are small enough w.r.t. the characteristic length, i.e., $\epsilon << 1$, then $\frac{\mathrm{St}}{\alpha+1} << 1$.

Overall, the mixture momentum conservation equation (8.16) becomes

$$\phi\frac{D_{\tilde{\mathbf{u}}_g}\tilde{\mathbf{u}}_g}{D_{\tilde{\mathbf{u}}_g}\tilde{t}} - \tilde{\nabla}\cdot\left[\phi\tilde{\boldsymbol{\sigma}}^{C}\right] = \phi\boldsymbol{e}_g,$$

which is the equation that we used for the dry case in Chapter 7 in the case of a vanishing Newtonian granular viscosity. However, in the vanishing grain size limit St will also get close to zero, and the flow will be incompressible, thus departing from the single phase equations.

Orders of magnitude We now look at the order of magnitude of the different terms in the dimensionless conservation equations for different physical parameters. Numbers of special interest are St^2 — the importance of the $\mathbf{w}$ transport term in the mixture momentum balance equation, $\frac{\mathrm{St}}{\mathrm{Re}} = \frac{\alpha\epsilon^2}{18}$ — the viscosity term on $\mathbf{w}$, and $\frac{\mathrm{St}}{\Xi}$ and $\frac{\mathrm{St}^2}{\Xi}$ – the inertial terms in the fluctuation equations.

Since as we already mentioned, we know that we overestimate $\mathbf{w}$, and since when using the Richardson-Zaki closure the term $\beta\Xi\tilde{\xi}$ will be greater that 1 for most regimes, $\mathrm{St} < 10^{-1}$ appears to be a satisfying criterion for neglecting all inertial forces in $\mathbf{w}$.

- We first consider sand grains in water, for which $\rho_f \sim 10^3\mathrm{kg.m}^{-3}$, $\rho_g \sim 2.5\times10^3\mathrm{kg.m}^{-3}$, and $\eta_f \sim 10^{-3}$Pa.s. We have $\mathrm{Re} = 3\times10^6 L^{\frac{3}{2}}$, $\alpha \sim 1$ and $\mathrm{St} = 10^5 L^{\frac{3}{2}}\epsilon^2 = 10^5 D_g^2 L^{-\frac{1}{2}}$. This scaling means that when the size of the particles doubles, the size of the domain has to be multiplied by 16 to recover the same ratio for the mass-averaged and fluctuation velocities, and by 256 to get a similar influence for the inertial terms in $\mathbf{w}$. For grains of diameter $D_g = 1$mm, getting $\mathrm{St} < 0.1$ requires $L > 1$m. However, for $D_g = 1$cm, we must have $L > 10$km. As $\frac{\mathrm{St}}{\mathrm{Re}} << 1$ as soon as $L > 10D_g$, the viscous forces on $\mathbf{w}$ can be neglected in both of those scenarios.

- Now, consider the same material in air, i.e., $\rho_f \sim 1\text{kg.m}^{-3}$, and $\eta_f \sim 10^{-3}\text{Pa.s}$. We have $\text{Re} = 3\times 10^5 L^{\frac{3}{2}}$, $\alpha \sim 2500$ and $\text{St} = 10^7 L^{\frac{3}{2}}\epsilon^2 = 10^7 D_g^2 L^{-\frac{1}{2}}$. For grains of diameter $D_g = 1\text{mm}$, getting $\text{St} < 0.1$ requires $L > 10^4\text{m}$. However, for $D_g = 100\mu\text{m}$, it suffices to have $L > 1\text{m}$. Moreover $\frac{\text{St}}{\text{Re}} << 1$ as soon as $L > 10^2 D_g$, thus in both cases the viscous forces on $\mathbf{w}$ can be neglected.

Note that these estimates are quite conservative, and that these values for the characteristic length can be reduced in regimes where the volume fraction of grains is not near zero.

Simplified model Now, supposing that we can neglect the inertial and viscous terms in $\mathbf{w}$, our conservation equations boil down to

$$\begin{cases} \dfrac{\beta}{\alpha+1}\dfrac{D\tilde{\mathbf{u}_m}}{D\tilde{t}} - \tilde{\nabla}\cdot\left[\phi\tilde{\boldsymbol{\sigma}}^C\right] - \dfrac{1}{\alpha+1}\tilde{\nabla}\cdot\left[\dfrac{\tilde{\eta_{\text{eff}}}}{\text{Re}}\tilde{\mathrm{D}}(\tilde{\mathbf{u}_m}) - \tilde{\mathrm{p}}\mathbb{I}\right] = \dfrac{\beta}{\alpha+1}\boldsymbol{e}_g. \\ \dfrac{\alpha\phi}{\alpha+1}\left(\tilde{\nabla}\tilde{\mathrm{p}} - \dfrac{1}{\text{Re}}\nabla\cdot\left[\tilde{\eta_{\text{eff}}}\tilde{\mathrm{D}}(\tilde{\mathbf{u}_m})\right]\right) + \tilde{\nabla}\cdot\left[\phi\tilde{\boldsymbol{\sigma}}^C\right] = \beta\phi\Xi\tilde{\xi}\tilde{\mathbf{w}}. \end{cases}$$

We are primarily interested in the grain velocity, which is given by $\tilde{\mathbf{u}}_g = \tilde{\mathbf{u}_m} + \tilde{\text{St}}\frac{\mathbf{w}}{\beta}$. The influence of the remaining viscous term in the simplified fluctuation conservation equation on $\mathbf{u}_g$ is thus of order $\frac{\text{St}}{\text{Re}}$, which we already assumed to be negligible. Our simplified model will therefore be defined as the solution to

$$\begin{cases} \dfrac{\beta}{\alpha+1}\dfrac{D\tilde{\mathbf{u}_m}}{D\tilde{t}} - \tilde{\nabla}\cdot\left[\phi\tilde{\boldsymbol{\sigma}}^C\right] - \dfrac{1}{\alpha+1}\tilde{\nabla}\cdot\left[\dfrac{\tilde{\eta_{\text{eff}}}}{\text{Re}}\tilde{\mathrm{D}}(\tilde{\mathbf{u}_m}) - \tilde{\mathrm{p}}\mathbb{I}\right] = \dfrac{\beta}{\alpha+1}\boldsymbol{e}_g \\ \dfrac{\alpha\phi}{\alpha+1}\tilde{\nabla}\tilde{\mathrm{p}} + \tilde{\nabla}\cdot\left[\phi\tilde{\boldsymbol{\sigma}}^C\right] = \dfrac{\alpha}{\alpha+1}\phi\beta\Xi\tilde{\xi}\tilde{\mathbf{w}} \\ \tilde{\nabla}\cdot\left[\tilde{\mathbf{u}_m} - \text{St}\alpha\dfrac{\pi}{\beta}\tilde{\mathbf{w}}\right] = 0 \\ \dfrac{\partial\beta}{\partial\tilde{t}} + \tilde{\nabla}\cdot[\beta\mathbf{u}_m] = 0, \end{cases}$$

with the supplemental condition that $\dot{\boldsymbol{\varepsilon}}_g = \mathrm{D}(\tilde{\mathbf{u}_m} + \text{St}\frac{1-\phi}{\beta}\tilde{\mathbf{w}})$ and $\tilde{\boldsymbol{\sigma}}^C$ should follow the Drucker–Prager flow rule of Equation (8.10). Since we have now an explicit expression for $\mathbf{w}$, we can eliminate this variable and get

$$\begin{cases} \dfrac{\beta}{\alpha+1}\dfrac{D\tilde{\mathbf{u}_m}}{D\tilde{t}} - \dfrac{1}{\alpha+1}\tilde{\nabla}\cdot\left[\dfrac{\tilde{\eta_{\text{eff}}}}{\text{Re}}\tilde{\mathrm{D}}(\tilde{\mathbf{u}_m}) - \tilde{\mathrm{p}}\mathbb{I}\right] + \tilde{\nabla}\cdot\left[\phi\tilde{\boldsymbol{\lambda}}\right] = \dfrac{\beta}{\alpha+1}\boldsymbol{e}_g \\ \dfrac{\tilde{\nabla}\cdot[\tilde{\mathbf{u}_m}]}{\alpha+1} - \dfrac{\text{St}}{\Xi}\tilde{\nabla}\cdot\left[\dfrac{\alpha}{\alpha+1}\dfrac{\pi}{\beta^2\xi}\tilde{\nabla}\tilde{\mathrm{p}} - \dfrac{1-\phi}{\beta^2\xi}\tilde{\nabla}\cdot\left[\phi\tilde{\boldsymbol{\lambda}}\right]\right] = 0 \\ \phi\tilde{\mathrm{D}}(\tilde{\mathbf{u}_m}) - \phi\dfrac{\text{St}}{\Xi}\tilde{\mathrm{D}}\left(\dfrac{1-\phi}{\beta^2\xi}\tilde{\nabla}\tilde{\mathrm{p}} - \dfrac{\alpha+1}{\alpha}\dfrac{1-\phi}{\phi\beta^2\xi}\tilde{\nabla}\cdot\left[\phi\tilde{\boldsymbol{\lambda}}\right]\right) = \tilde{\boldsymbol{\gamma}}, \end{cases} \tag{8.21}$$

with once again our usual frictional rheology on $\boldsymbol{\gamma} := \phi\,\mathrm{D}(\mathbf{u}_f)$ and $\boldsymbol{\lambda} := -\boldsymbol{\sigma}^C$. Each row in (8.21) as been scaled so as to highlight the symmetry of the system.

This model is interesting for a few reasons:

- We still retrieve an incompressible flow for $\text{St} = 0$ or $\Xi = \infty$;
- While this does not fall *at all* in the range of validity of our hypothesis, we also retrieve the dry granular equations of Chapter 7 when taking the $\alpha = \infty$, $\text{St} = \infty$ limit;
- Ignoring the grain–grain contact stress, the fluctuation equations now reads exactly as Darcy's law;

- Still ignoring grain-grain interactions, the system we have to solve is deeply similar to that of near-incompressible flows, i.e., a penalized Navier-Stokes problem,

$$\begin{cases} \dfrac{\beta}{\alpha+1}\dfrac{D\tilde{\mathbf{u}_m}}{D\tilde{t}} - \dfrac{1}{\alpha+1}\tilde{\nabla}\cdot\left[\dfrac{\tilde{\eta_{\text{eff}}}}{\text{Re}}\tilde{\text{D}}(\tilde{\mathbf{u}_m}) - \tilde{p}\mathbb{I}\right] = \dfrac{\beta}{\alpha+1}\boldsymbol{e}_g \\ \dfrac{\tilde{\nabla}\cdot[\tilde{\mathbf{u}_m}]}{\alpha+1} - \dfrac{\alpha}{\alpha+1}\dfrac{\text{St}}{\Xi}\tilde{\nabla}\cdot\left[\dfrac{\pi}{\beta^2\xi}\tilde{\nabla}\tilde{p}\right] = 0 \\ \dfrac{\partial\beta}{\partial\tilde{t}} + \tilde{\nabla}\cdot[\beta\mathbf{u}_m] = 0. \end{cases}$$

The reduced diphasic model (8.21) is very similar to the single phase model of Chapter 7, except that the degrees of freedom are now the combination of the mass-averaged velocity $\mathbf{u}_m$ *and* the pressure p. It could be solved in a similar fashion, but one would have to be wary that the stiffness matrix is no longer guaranteed to be positive-definite, and that another term, coming from the nonlinear Laplacian of $\boldsymbol{\lambda}$ in the definition of $\boldsymbol{\gamma}$, contributes to the Delassus operator. Since $\mathbf{w}$ does not need to be discretized, this reduced problem could be cheaper to solve than the original one, and may still be able to capture a wide range of phenomena. Following the approach of the dry case, the simplified model can also be shown to be dissipative. Despite these potential advantages, in the following section we will go back to the general case and propose a numerical method to solve our original equations. Adapting this method to the simplified model should be straightforward.

8.3 Numerical resolution of the two-phase equations

We consider our original dimensionless system of four equations — the two conservation equations, (8.16) and (8.17), and the two mass conservation equations (8.18) and (8.20) — plus our frictional rheology constraint (8.10). We now drop the tildes to lighten notations.

8.3.1 Time discretization

Just like in the dry case presented in Chapter 7, we will use a semi-implicit timestepping scheme, that first solves the momentum conservation equations, then advect the volume fraction field. That is, we will first solve equations (8.16), (8.17) subject to the volume-averaged incompressibility (8.18) and the frictional rheology (8.10) to get end-of-step values for $\mathbf{u}$,$\mathbf{w}$, p and $\boldsymbol{\sigma}^C$, then solve the volume fraction transport equation (8.20) using these new velocities.

Frictional contacts Once again, we thus have to linearize the end-of-step maximal volume fraction constraint, following

$$(\phi(t+\Delta_t) \le \phi_{\max}) \sim (\phi^k + \phi^k \Delta_t \,\text{D}(\mathbf{u}_g) \le \phi_{\max}).$$

Just as in Chapter 7, we thus define

$$\boldsymbol{\gamma} := \phi^k \,\text{D}(\mathbf{u}_g) + \frac{\phi_{\max} - \phi}{\Delta_t}\frac{\mathbb{I}}{d} \qquad \text{and } \boldsymbol{\lambda} := -\boldsymbol{\sigma}^C, \tag{8.22}$$

such that our frictional rheology constraint is expressed as $(\boldsymbol{\gamma}, \lambda) \in \mathscr{D}\mathscr{P}(\mu)$.

Material velocity derivatives We suppose that the material derivative for the velocity of each phase can be approximated to the first order as

$$\frac{D_{\mathbf{u}_{f,g}}\mathbf{u}_{f,g}}{D_{\mathbf{u}_{f,g}}t} = \frac{\mathbf{u}_{f,g}^{k+1} - u_{f,g}(\mathbf{u}_{f,g}^k)}{\Delta_t} + O(\Delta_t).$$

For instance, u can be computed using the characteristics morphism, or by recovering the velocity from particles in the case of hybrid schemes.

Using this approximation, we can get simpler expressions for the inertial terms of the mixture and fluctuation momentum conservation equations. Indeed,

$$\phi(\alpha+1)\frac{D_{\mathbf{u}_g}\mathbf{u}_g}{D_{\mathbf{u}_g}t}+(1-\phi)\frac{D_{\mathbf{u}_f}\mathbf{u}_f}{D_{\mathbf{u}_f}t}\sim\frac{1}{\Delta_t}\Big[\beta\mathbf{u}_{\mathrm{m}}-(\alpha+1)\phi u_g(\mathbf{u}_g^k)-(1-\phi)u_f(\mathbf{u}_f^k)\Big]$$
$$\frac{D_{\mathbf{u}_g}\mathbf{u}_g}{D_{\mathbf{u}_g}t}-\frac{D_{\mathbf{u}_f}\mathbf{u}_f}{D_{\mathbf{u}_f}t}\sim\frac{1}{\Delta_t}\Big[\mathrm{St}\,\mathbf{w}+u_f(\mathbf{u}_f^k)-u_g(\mathbf{u}_g^k)\Big].$$

Using this insight, we rewrite our momentum conservation equations as

$$\frac{\beta}{\alpha+1}\frac{\mathbf{u}_{\mathrm{m}}}{\Delta_t}+\nabla\cdot[\phi\boldsymbol{\lambda}]-\frac{1}{\alpha+1}\nabla\cdot\left[\frac{\eta_{\mathrm{eff}}}{\mathrm{Re}}\mathrm{D}(\mathbf{u}_{\mathrm{m}}-\mathrm{St}\alpha\frac{\pi}{\beta}\mathbf{w})-\tilde{\mathrm{p}}\mathbb{I}\right]$$
$$=\frac{\beta}{\alpha+1}\boldsymbol{e}_g+\frac{1}{\Delta_t}\left[\phi u_g(\mathbf{u}_g^k)+\frac{1-\phi}{\alpha+1}u_f(\mathbf{u}_f^k)\right]\quad(8.23)$$

$$\phi\left(\frac{\mathrm{St}}{\Delta_t}+\frac{\alpha\beta}{\alpha+1}\Xi\xi\right)\mathbf{w}-\frac{\alpha\phi}{\alpha+1}\left(\nabla\mathrm{p}-\frac{1}{\mathrm{Re}}\nabla\cdot\left[\eta_{\mathrm{eff}}\mathrm{D}(\mathbf{u}_{\mathrm{m}}-\alpha\mathrm{St}\frac{\pi}{\beta}\mathbf{w})\right]\right)$$
$$+\nabla\cdot[\phi\boldsymbol{\lambda}]=\frac{\phi}{\Delta_t}\Big(u_g(\mathbf{u}_g^k)-u_f(\mathbf{u}_f^k)\Big)\quad(8.24)$$

8.3.2 Variational formulation

We first perform a change of variable which will allow us to obtain a symmetric system, and introduce the velocity field $\hat{\mathbf{w}}:=\sqrt{\mathrm{St}}\frac{1-\phi}{\beta}\mathbf{w}$. The velocity of each phase can be reconstructed from $\mathbf{u}_{\mathrm{m}}$ and $\hat{\mathbf{w}}$ as $\mathbf{u}_g=\mathbf{u}_{\mathrm{m}}+\sqrt{\mathrm{St}}\hat{\mathbf{w}}$ and $\mathbf{u}_f=\mathbf{u}_{\mathrm{m}}-(\alpha+1)\sqrt{\mathrm{St}}\frac{\phi}{1-\phi}\hat{\mathbf{w}}$.

Let us consider a simulation domain Ω, with, for the sake of simplicity, homogeneous Dirichlet boundary conditions. Let as usual V denote a subspace of $H^1(\Omega)^d$ satisfying the boundary conditions, and $T(\Omega)\sim L_2(\Omega)^{s_d}$ be the space of square-integrable symmetric tensor fields on Ω. Our two-phase flow is described by the solution to the following variational formulation:

Find $\mathbf{u}_{\mathrm{m}},\hat{\mathbf{w}}\in V^2$, $\mathrm{p}\in L_2(\Omega)$ and $\boldsymbol{\gamma},\boldsymbol{\lambda}\in T^2$ such that

$$\begin{array}{rcrcrcrcll}
a(\mathbf{u}_{\mathrm{m}},\mathbf{v}) & + & e(\hat{\mathbf{w}},\mathbf{v}) & - & b(\mathrm{p},\mathbf{v}) & - & g(\boldsymbol{\lambda},\mathbf{v}) & = & l(\mathbf{v}) & \forall\mathbf{v}\in V\\
e(\mathbf{u}_{\mathrm{m}},\mathbf{z}) & + & r(\hat{\mathbf{w}},\mathbf{z}) & - & c(\mathrm{p},\mathbf{z}) & - & h(\boldsymbol{\lambda},\mathbf{z}) & = & f(\mathbf{z}) & \forall\mathbf{z}\in V\\
-b(\mathbf{q},\mathbf{u}_{\mathrm{m}}) & - & c(\mathbf{q},\hat{\mathbf{w}}) & & & & & = & 0 & \forall\mathbf{q}\in L_2(\Omega)\\
-g(\boldsymbol{\tau},\mathbf{u}_{\mathrm{m}}) & - & h(\boldsymbol{\tau},\hat{\mathbf{w}}) & & & + & m(\boldsymbol{\gamma},\boldsymbol{\tau}) & = & k(\boldsymbol{\tau}) & \forall\boldsymbol{\tau}\in T\\
& & & & & & (\boldsymbol{\gamma},\boldsymbol{\lambda}) & \in & \mathscr{DP}(\mu). &
\end{array}$$

where $a=a_1+a_2$, $r=r_1+r_2$,

$$a_1(\mathbf{u},\mathbf{v}):=\frac{1}{\Delta_t}\int_\Omega\frac{\beta}{\alpha+1}\langle\mathbf{u},\mathbf{v}\rangle\qquad a_2(\mathbf{u},\mathbf{v}):=\frac{1}{(\alpha+1)\mathrm{Re}}\int_\Omega\eta_{\mathrm{eff}}\mathrm{D}(\mathbf{u}):\mathrm{D}(\mathbf{v})$$
$$r_1(\mathbf{w},\mathbf{z}):=\int_\Omega\frac{\phi\beta}{1-\phi}\left(\frac{\mathrm{St}}{\Delta_t}+\frac{\alpha\beta}{\alpha+1}\Xi\xi\right)\qquad r_2(\mathbf{w},\mathbf{z}):=\frac{\mathrm{St}}{\mathrm{Re}}\frac{\alpha^2}{\alpha+1}\int_\Omega\eta_{\mathrm{eff}}\mathrm{D}(\phi\mathbf{w}):\mathrm{D}(\phi\mathbf{z})$$
$$e(\mathbf{w},\mathbf{z}):=-\frac{\alpha\sqrt{\mathrm{St}}}{\alpha+1}\int_\Omega\frac{\eta_{\mathrm{eff}}}{\mathrm{Re}}\mathrm{D}(\phi\mathbf{w}):\mathrm{D}(\mathbf{v})\qquad m(\boldsymbol{\gamma},\boldsymbol{\tau}):=\int_\Omega\boldsymbol{\gamma}:\boldsymbol{\tau}$$
$$b(\mathrm{p},\mathbf{v}):=\frac{1}{\alpha+1}\int_\Omega\mathrm{p}\nabla\cdot\mathbf{z}\qquad c(\mathrm{p},\mathbf{z}):=\frac{\alpha\sqrt{\mathrm{St}}}{\alpha+1}\int_\Omega\phi\langle\nabla\mathrm{p},\mathbf{z}\rangle$$
$$g(\boldsymbol{\tau},\mathbf{u}):=\int_\Omega\phi\boldsymbol{\tau}:\mathrm{D}(\mathbf{u})\qquad h(\boldsymbol{\tau},\mathbf{u}):=\sqrt{\mathrm{St}}\int_\Omega\phi\,\boldsymbol{\tau}:\mathrm{D}(\mathbf{w})$$

$$
\begin{aligned}
l(\mathbf{v}) &:= \int_\Omega \frac{\beta}{\alpha+1}\langle \boldsymbol{e}_g, \mathbf{v}\rangle + \frac{1}{\Delta_t}\int_\Omega \Big\langle (\alpha+1)\phi u_g(\mathbf{u}_g^k) + (1-\phi)u_f(\mathbf{u}_f^k), \mathbf{v}\Big\rangle \\
k(\boldsymbol{\tau}) &:= \int_\Omega \frac{\phi_{\max}-\phi}{\Delta_t}\frac{\mathrm{Tr}\,\boldsymbol{\tau}}{d} \\
f(\mathbf{z}) &:= \frac{\sqrt{\mathrm{St}}}{\Delta_t}\int_\Omega \phi\Big\langle u_g(\mathbf{u}_g^k) - u_f(\mathbf{u}_f^k), \mathbf{z}\Big\rangle.
\end{aligned}
$$

Proof. Scaling the fluctuation momentum conservation equation (8.24) by $\sqrt{\mathrm{St}}$, dividing the volume-averaged incompressibility (8.18) by $(\alpha+1)$ and transcribing the definition of $\hat{\mathbf{w}}$ into them, we obtain:

$$
\begin{aligned}
\frac{\phi\beta}{1-\phi}\left(\frac{\mathrm{St}}{\Delta_t}+\frac{\alpha\beta}{\alpha+1}\Xi\xi\right)\hat{\mathbf{w}} - \frac{\alpha\phi}{\alpha+1}\left(\sqrt{\mathrm{St}}\nabla \mathrm{p} - \frac{1}{\mathrm{Re}}\nabla\cdot\left[\eta_{\mathrm{eff}}\mathrm{D}(\sqrt{\mathrm{St}}\mathbf{u}_{\mathrm{m}} - \alpha\mathrm{St}\phi\hat{\mathbf{w}})\right]\right) \\
+ \sqrt{\mathrm{St}}\nabla\cdot[\phi\boldsymbol{\lambda}] = \sqrt{\mathrm{St}}\frac{\phi}{\Delta_t}\Big(\mathbf{u}_g(\mathbf{u}_g^k) - \mathbf{u}_f(\mathbf{u}_f^k)\Big),
\end{aligned}
$$

$$
\frac{1}{\alpha+1}\nabla\cdot\mathbf{u}_{\mathrm{m}} - \sqrt{\mathrm{St}}\frac{\alpha}{\alpha+1}\nabla\cdot[\phi\hat{\mathbf{w}}] = 0,
$$

which, put under weak form and using Green formulas, correspond to the second and third lines of our variational formulation. The first and fourth line are obtained through a similar process on the mixture momentum conservation equation (8.24) and the definition of $\boldsymbol{\gamma}$, Equation (8.22). □

8.3.3 Discrete system

Choosing adequate discrete spaces for our variables, and expressing the constraints on quadrature points as we did in Chapters 6 and (7), we ultimately obtain a system which has the form:

$$
\begin{cases}
A\underline{\boldsymbol{u}} + E^{\intercal}\underline{\boldsymbol{w}} = \underline{\boldsymbol{l}} + B^{\intercal}\underline{\boldsymbol{p}} + G^{\intercal}\underline{\boldsymbol{\lambda}} \\
E\underline{\boldsymbol{u}} + R\underline{\boldsymbol{w}} = \underline{\boldsymbol{f}} + C^{\intercal}\underline{\boldsymbol{p}} + H^{\intercal}\underline{\boldsymbol{\lambda}} \\
\mathbf{0} = B\underline{\boldsymbol{u}} + C\underline{\boldsymbol{w}} \\
\underline{\boldsymbol{\gamma}} = \underline{\boldsymbol{k}} + G\underline{\boldsymbol{u}} + H\underline{\boldsymbol{w}} \\
(\underline{\boldsymbol{\lambda}}_{[i]}, \underline{\boldsymbol{\gamma}}_{[i]}) \in \mathscr{DP}(\mu) \quad \forall i.
\end{cases}
\tag{8.25}
$$

System (8.25) is a DCFP, with linear constraints and whose degrees of freedom are given by the couple $(\underline{\boldsymbol{u}}, \underline{\boldsymbol{w}})$. Concatenating the system matrices as

$$
\bar{A} := \begin{pmatrix} A & E^{\intercal} \\ E & R \end{pmatrix} \qquad \bar{H} := \begin{pmatrix} G, H \end{pmatrix} \qquad \bar{C} := \begin{pmatrix} B, C \end{pmatrix},
$$

System (8.25) can be rewritten in a more familiar manner as

$$
\begin{cases}
\bar{A}\begin{pmatrix}\underline{\boldsymbol{u}}; \underline{\boldsymbol{w}}\end{pmatrix} = \begin{pmatrix}\underline{\boldsymbol{l}} \\ \underline{\boldsymbol{f}}\end{pmatrix} + \bar{C}^{\intercal}\underline{\boldsymbol{p}} + \bar{H}^{\intercal}\underline{\boldsymbol{\lambda}} \\
\bar{C}\begin{pmatrix}\underline{\boldsymbol{u}}; \underline{\boldsymbol{w}}\end{pmatrix} = \mathbf{0} \\
\underline{\boldsymbol{\gamma}} = \underline{\boldsymbol{k}} + \bar{H}\begin{pmatrix}\underline{\boldsymbol{u}}; \underline{\boldsymbol{w}}\end{pmatrix} \\
(\underline{\boldsymbol{\lambda}}_{[i]}, \underline{\boldsymbol{\gamma}}_{[i]}) \in \mathscr{DP}(\mu) \quad \forall i.
\end{cases}
\tag{8.26}
$$

The matrix A is always symmetric positive-definite, and discretizing the fluctuation field only where $\hat{\phi} > 0$ — which is sensible, as $\hat{\mathbf{w}}$ should vanish elsewhere — R is also positive-definite; therefore $\bar{A}$ is symmetric positive-positive as well.

Numerical resolution Chapters 3 and 4 were dedicated to solving numerically the DCFP with linear constraints. However, notice that System 8.26 features two peculiarities that restrict the choice of numerical solvers.

- First, the inverse of the stiffness matrix, $\bar{A}^{-1}$, is dense. This difficulty can be dealt with by either using the numerical algorithms proposed in Chapter 6, or making the high Reynolds hypothesis of Chapter 7 and using the two-steps algorithm of Section 7.3.4.
- The matrix $\bar{C}$ is not surjective — the pressure field is simply defined up to a constant — and thus the Schur complement of the linear constraints may not be invertible, discarding some of the methods that we proposed to deal with linear constraints in a DCFP. If the discretization spaces for the velocities and pressure, V_h and Q_h, satisfy an *inf-sup* criterion, this problem can be solved by enforcing a supplemental zero-average constraint for the pressure p. Otherwise, we can assume that slightly relaxing the incompressibility constraint of the volume-averaged velocity is acceptable, and penalize the pressure variable.

In practice, as we put more of an emphasis on solving efficiency rather than on the precision of the fluid velocity field, we use the two-step approximate algorithm together with the relaxation of the incompressibility constraints.

That is, we first solve a Stokes-like problem, with either the zero-average constraint or a penalization strategy, depending on whether our choice for V_h and Q_h satisfies the *inf-sup* condition, to obtain a candidate velocity $\left(\underline{\boldsymbol{u}};\underline{\boldsymbol{w}}\right)^*$, e.g.,

$$\begin{pmatrix} A & E^\intercal & -B^T & \\ E & R & -C^T & \\ -B & -C & & \mathbb{1} \\ & & \mathbb{1}^\intercal & \end{pmatrix}\begin{pmatrix} \underline{\boldsymbol{u}}^* \\ \underline{\boldsymbol{w}}^* \\ \underline{\boldsymbol{p}} \\ \kappa \end{pmatrix} = \begin{pmatrix} \underline{\boldsymbol{l}} \\ \underline{\boldsymbol{f}} \\ \underline{\mathbf{0}} \\ 0 \end{pmatrix} \quad \text{or} \quad \begin{pmatrix} A & E^\intercal & -B^T \\ E & R & -C^T \\ -B & -C & -c\mathbb{I} \end{pmatrix}\begin{pmatrix} \underline{\boldsymbol{u}}^* \\ \underline{\boldsymbol{w}}^* \\ \underline{\boldsymbol{p}} \end{pmatrix} = \begin{pmatrix} \underline{\boldsymbol{l}} \\ \underline{\boldsymbol{f}} \\ \underline{\mathbf{0}} \end{pmatrix},$$

where κ is the Lagrange multiplier associated to the zero-pressure average condition.

Then, we solve a lumped DCFP to get the velocity increment satisfying the frictional constraints,

$$\begin{cases} \tilde{\bar{A}}\Delta\left(\underline{\boldsymbol{u}};\underline{\boldsymbol{w}}\right) = \bar{C}^\intercal\underline{\boldsymbol{p}} + \bar{H}^\intercal\underline{\boldsymbol{\lambda}} \\ \bar{C}\Delta\left(\underline{\boldsymbol{u}};\underline{\boldsymbol{w}}\right) = -c\underline{\boldsymbol{p}} \\ \qquad\qquad \underline{\boldsymbol{\gamma}} = \bar{H}\left(\underline{\boldsymbol{u}};\underline{\boldsymbol{w}}\right) + \underline{\boldsymbol{k}} + \bar{H}\left(\underline{\boldsymbol{u}};\underline{\boldsymbol{w}}\right)^* \\ \left(\underline{\boldsymbol{\lambda}}_{[i]},\underline{\boldsymbol{\gamma}}_{[i]}\right) \in \mathscr{D}\mathscr{P}(\mu) \quad \forall i, \end{cases} \tag{8.27}$$

where $\tilde{\bar{A}} = \mathrm{diag}(\tilde{A}_1, \tilde{R}_1)$ is the concatenation of the lumped mass matrices for the mixture and fluctuation inertia. In practice, we use the matrix-free Gauss–Seidel algorithm presented in Chapter 4, where the linear constraints are accounted for every few iterations by solving the symmetric positive-definite system

$$(\bar{C}\tilde{\bar{A}}^{-1}\bar{C}^\intercal + c\mathbb{I})\underline{\boldsymbol{p}} = -\bar{C}\tilde{\bar{A}}^{-1}\bar{H}^T\underline{\boldsymbol{\lambda}}^k.$$

8.3.4 Spatial discretization

In the previous paragraph we assumed that suitable discretization spaces for the velocity, pressure, grain volume fraction and stress fields had been chosen. We did not pursue this investigation in details yet, and simply adapted the discretization strategies devised for the dense case. We considered either:

- Piecewise-constant approximations on a triangular mesh for the velocity, volume fraction, and stress field, with a piecewise-linear pressure field. As in Section 7.2.2, advection of the volume fraction and velocity fields was done using an upwind scheme.
- MPM discretization of the volume fraction field, as in Section 7.3, with trilinear velocities and trilinear, trilinear-discontinuous, or particle-based stresses. For the sake of simplicity

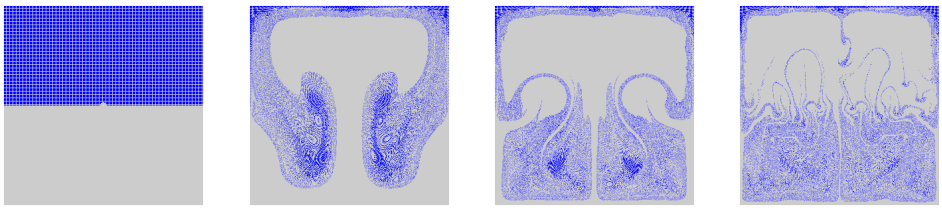


Figure 8.2: *Rayleigh–Taylor instability. A slight perturbation is introduced at the center of the interface between an upper, heavy fluid and a lighter one (here $\alpha = 1000$). The fluids are supposed fully immiscible ($St = 0$) and viscous ($Re = 3.\ 10^{-2}$).*

we also used trilinear pressure field, which has very poor theoretical properties yet yielded satisfying results when relaxing the volume-incompressibility constraint. The advected velocity of the granular phase is recovered from the particles, while the characteristics method is used for the fluid velocities. Particle-based advection could also be used for the fluid, but one would then have to perform resampling to ensure that the sum of volume fractions of both phases remain always equal to 1.

8.4 Results

All the results presented below were obtained with an extension of our MPM simulation framework from Chapter 7 to diphasic flows.

8.4.1 Rayleigh-Taylor instability

We performed a first test to check that independently of our treatment of the granular rheology, our diphasic simulation code was behaving as expected. Figure 8.2 shows the development of a Rayleigh–Taylor instability between two Newtonian fluids, the upper one being much heavier than the bottom one. We capture the characteristic plumes associated to this instability (compare with a similar simulation in e.g., Boyer 2001, Figure II.4.12), though our particle-based discretization leads to a rather messy end state.

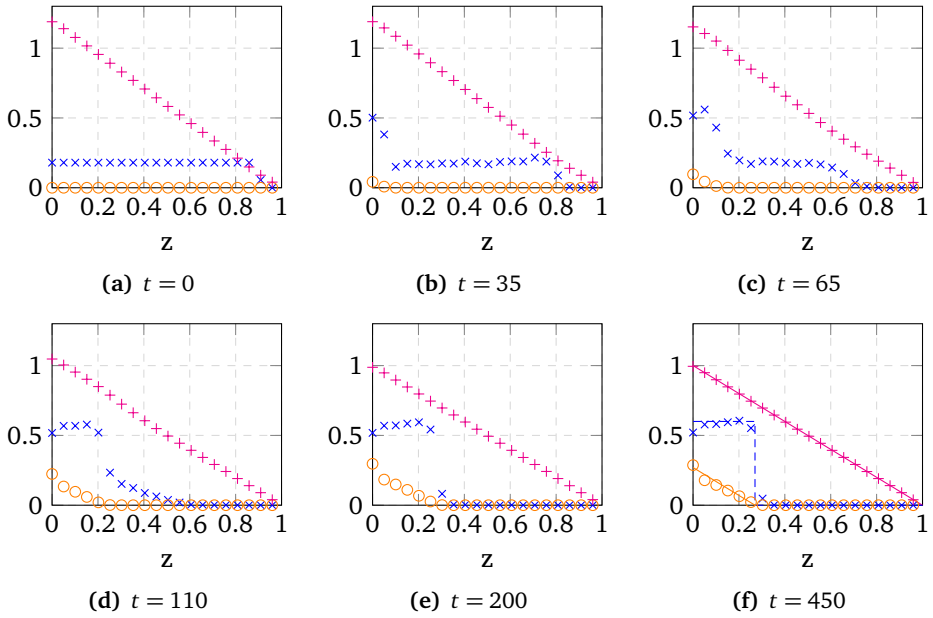
8.4.2 Sedimentation

We consider a 2D water container of width and height $W = H = 1$m, rigid grains of volumetric mass $\rho_g = 2500\text{kg.m}^{-3}$, and a maximal volume fraction $\phi_{\max} = 0.6$. The initial volume fraction of grains $\phi^0(x,z)$ is such that

$$\phi^0(x,z) = \begin{cases} 0.3\phi_{\max} & \text{if } z \leq H_0 = 0.9H \\ 0 & \text{if } z > H_0. \end{cases}$$

Figure 8.3 shows the evolution of the volume fraction, pore pressure p and particulate normal stress $\text{Tr}\,\boldsymbol{\lambda}$ fields averaged over the width of the container for grains of diameter $D_g = 0.5$mm. Following the considerations of Section 8.2.6, the pressure has been made dimensionless w.r.t. the characteristic pressure $\rho_f g H$, and the particulate normal stress w.r.t. $\alpha\rho_f g H$. We observe that the final state of our simulated system does correspond to the analytical predictions of Section 8.2.6.

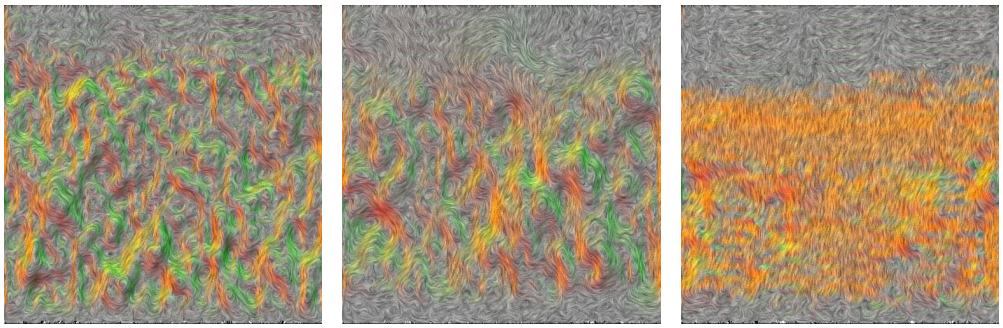
Influence of grain diameters The diameter of the grains (or rather, the ratio of length scales $\epsilon := D_g/H$) has a dramatic influence on the sedimentation process. However, the grains diameters has no influence on the final state of the system.



(a) $t = 0$ **(b)** $t = 35$ **(c)** $t = 65$

(d) $t = 110$ **(e)** $t = 200$ **(f)** $t = 450$

Figure 8.3: *Horizontally-averaged volume fraction of grains, dimensionless pore pressure and effective stress at different instants of a simulated sedimentation process. In the lower-right graph, lines correspond to the theoretical values from Section 8.2.6.*



(a) $\epsilon = 3.10^{-4}$ **(b)** $\epsilon = 5.10^{-4}$ **(c)** $\epsilon = 10^{-3}$

Figure 8.4: *Line-integral convolution of the velocity field at the onset of the sedimentation process for different length ratios $\epsilon = D_g/H$. Hue indicates direction, brightness amplitude.*

For grains of diameter 0.3mm, the dimensionless duration of the sedimentation process (scaled by $H/\sqrt{\frac{\alpha}{\alpha+1}gH}$) was 30. It dropped to 17 for $D_g = 0.5$mm and to 6.5 for $D_g = 5$mm. This is explained by a much more turbulent velocity field for lower grain diameters, as illustrated in Figure 8.4.

8.4.3 Regimes

We attempted to reproduce the immersed column collapses of Topin, Monerie, et al. (2012). [4]

[4] http://www.irsn.fr/EN/Research/publications-documentation/Publications/PSN-RES/Pages/2012-Topin-collapse-dynamics-run-out-dense-granular-materials-fluid-videos.aspx,

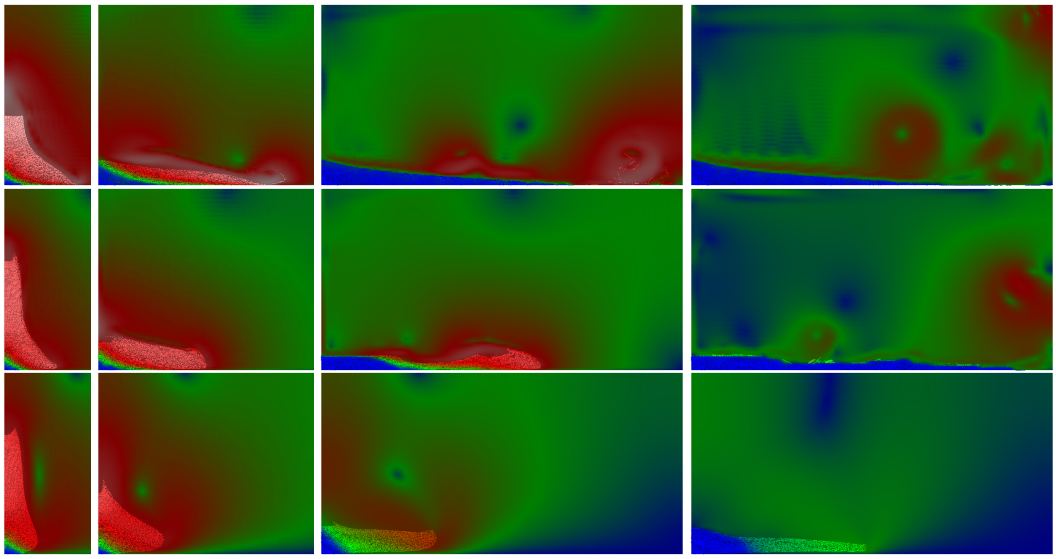


Figure 8.5: *Snapshots at identical instants of the collapse of a granular column in three different fluids: air (top), water (middle), and viscous (bottom). Colors indicate the particle and fluid velocities, on the same scale (blue is slowest, white fastest).*

We consider a 2D granular column of aspect-ratio $a = 8$, with grains of diameter $D_g = 1$mm and volumetric mass $\rho_g = 2600\text{kg.m}^{-3}$. The column width is $W = 11.5D_g$. Topin, Monerie, et al. (2012) couple the NSCD method with a fluid simulation; we use instead the $\mu(I)$ rheology to model the frictional contacts, with $\mu_S = 0.32$, $\mu_D = 0.6$. Three choices are considered for the surrounding fluid:

1. *air*: $\rho_f = 1\text{kg.m}^{-3}$ and $\eta_f = 10^{-5}$Pa.s;
2. *water*: $\rho_f = 1000\text{kg.m}^{-3}$ and $\eta_f = 10^{-3}$Pa.s;
3. *viscous*: $\rho_f = 1000\text{kg.m}^{-3}$ and $\eta_f = 1$Pa.s.

Topin, Monerie, et al. (2012) observe that each of these choices yield a different collapse regime, which they coin respectively *grain-inertial*, *fluid-inertial*, and *viscous*. In the latter case, the collapse is simply slowed-down by the fluid, and the run-out length is much shorter than in the dry (i.e., grain-inertial) case. However, in the fluid-inertial regime, the kinetic energy initially transferred from the grains to the fluid is transferred back to the grains in the later stage of the collapse, maintaining an horizontal velocity for a much longer time than in the two other regimes. As such, the final run-out length in the fluid-inertial regime can surpass that of the dry case.

While our experiments, reproduced in Figure 8.5, allow us to retrieve the grain-inertial and fluid-inertial regimes for the choices of parameters (1.) and (2.), choice (3.) remains mostly in the fluid-inertial regime; albeit much slower than the collapse in water, the collapse in the viscous fluid achieve a barely shorter run-out length. While initially surprising, this can be explained by looking at the characteristic numbers of our model.

Indeed, in the later stage of the collapse, the grains can be either slowed down or driven by the drag forces, depending on which of the contact stress or pore pressure is preponderant. The total normal stress of the mixture is $\mathrm{p} + \mathrm{Tr}\,\phi\boldsymbol{\lambda}$; the fact that the velocity is mostly horizontal, and thus that the grains are not collapsing under their own weight, means that the total normal stress is higher there than in the front of the avalanche. When the grains are weakly contacting, and

http://www.irsn.fr/EN/Research/publications-documentation/Publications/PSN-RES/Pages/2012-Topin-collapse-dynamics-run-out-dense-granular-materials-fluid-videos-bis.aspx

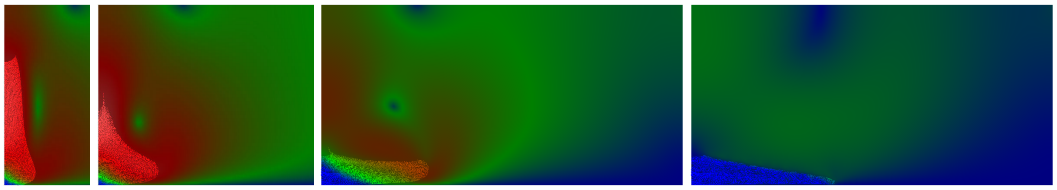


Figure 8.6: *Granular column collapse in the viscous (bottom) fluid of Figure 8.5 with triple-sized grains.*

thus the weight of the mixture is mostly supported by the pore pressure p, the relative velocity $\mathbf{w}$ is along ∇p. The drag force on the grains $\xi\mathbf{w}$ will be oriented towards the front of the flow, and will sustain it — this is the flow-inertial regime. On the other hand, when the weight of the granular phase is supported by grain–grain forces, $\mathbf{w}$ will be along $-\nabla \operatorname{Tr} \phi\boldsymbol{\lambda}$, and the drag forces will tend to slow down the grains — this is the viscous regime.

To determine which regime will drive the flow, we can thus look at the ratio between two timescales: the one for the viscous collapse, $T_v := \eta_f/\rho_f g L$, where L is the height of the basin, and the one for the grains to come into contact, $T_c := D_g/W$. We have

$$\frac{T_v}{T_c} = \frac{\eta_f W}{\rho_f g L D_g} = \frac{\eta_f \mathrm{St} U}{\rho_f U^2 L \epsilon} = \frac{\mathrm{St}}{\mathrm{Re}\epsilon} = \frac{\alpha+1}{18}\epsilon.$$

More than the Reynolds number Re, the $(\alpha+1)\epsilon$ will thus determine the regime of the collapse; the fluid-inertial one will occur when T_v/T_c is sufficiently small. For our column, $\epsilon = 10^{-2}$; in the dry case, $\alpha = 2599$ and thus $T_v/T_c \sim 1$; the collapse will therefore be quickly stabilized by the contact forces. However, this number remains constant for the choice of parameters (2.) and (3.), $\alpha\epsilon \sim 10^{-3}$. One solution to model a viscous collapse is thus to increase ϵ, i.e., consider larger grains. With our simulator, starting from the choice of parameters (3.) and simply tripling the diameters of the particles allows us to significantly reduce the duration and run-out length of the collapse (Figure 8.6).

Discussion

8.4.4 Limitations

It appears that our modeling of the drag term in the interfacial momentum transfer term is insufficient to retrieve the transition between the fluid-inertial and viscous regimes predicted by the discrete simulations. A nonlinear dependency of the drag force on the fluid viscosity for higher concentrations of particles might be necessary.

Another ingredient missing from our model is a dependency on the initial volume fraction of grains, which Pailha et al. (2008) showed had a tremendous influence on the onset of the flow. However, note that when using the Richardson–Zaki drag model, a higher volume fraction will mean that the fluid will oppose more resistance to the initial motion of the grains, ans thus will make stronger compact granular heaps.

Our approach is also intrinsically limited to fully immersed granular materials, and thus cannot be applied to the simulation of grains–air–water mixtures, such as wet sand. Finally, our method is not adequate for the simulation of phenomena which happen at a time scale much longer than that of the transport of individual grains, such as the transport of dunes by the wind.

On the numerical side, our diphasic simulation framework obviously suffers from all the discretization issues of the dry case discussed in Chapter 7, with the additional difficulty of having to find an adequate space for the discretization of the pore pressure field — which we sidestep in practice by relaxing the volume-average incompressibility constraint.

8.4.5 Conclusion

Despite all those drawbacks, our approach is still able to recover the qualitative dynamics of immersed granular flows in different regimes. We are able to capture the duality of the fluid role, which may either lubricate the flow or dampen it, following the relative importance of the pore pressure and contact forces. Moreover, we were able to cast the numerical problem that must be solved at each timestep as a standard DCFP with linear constraints, allowing us to leverage once again the large body of research devoted to such systems.

Future work will be focused on better understanding the role of the drag function ξ, which discriminates between the inertial and viscous regime. Moreover, as already mentioned for the dry case, the search for better discretization spaces constitutes an important research direction for us.

Conclusion

This dissertation focused on the simulation of complex materials featuring a large number of inelastic contacts with friction. Our main argument is that dealing with these contacts using methods derived from nonsmooth analysis is an efficient alternative to seemingly simpler elasticity- or regularization-based strategies. Indeed, nonsmooth methods allow to decouple the time scale of the macroscopic phenomenon from that of the contact dynamics themselves, the latter being often much smaller for stiff materials and of little practical interest. For instance, one avoids having to explicitly simulate the compression and decompression phases after an impact, and can thus get away with much larger timesteps.

We have also seen that continuum-based simulation methods can provide an efficient manner of simulating very large systems with small inhomogeneities, while still allowing to capture qualitative flow characteristics. Moreover, some numerical methods that have been developed for discrete mechanics can be translated in a straightforward manner to continuum materials. In particular, granular materials may be simulated either at the grain level with the Coulomb friction law or in a macroscopic fashion using the non-associated Drucker–Prager flow rule. Both approaches can leverage the exact same numerical solvers — even though the latter constitutive law is expressed on tensorial rather than vectorial quantities. This legitimizes further the focus that we have put on devising a robust an efficient solver for DCFP.

Finally, we have advocated in very different settings the use of *hybrid* methods, which can take insight from distinct point of views to alleviate the drawbacks of each of them. We have highlighted the benefits of combining an optimization-based approach with an analytical one for solving local problems arising when using splitting methods on DCFP. Similarly, we have seen that using simultaneously mesh-based and particle-based discretization strategies can simplify the treatment of both temporal and spatial differential operators.

9.1 Key remarks and summary of contributions

We recall below with more details the main points that we brought up over the course of this thesis, and emphasize the original contributions that we have introduced.

Following the works of Alart and de Saxcé, Chapter 1 presented the Signorini–Coulomb frictional contact law and the non-associated Drucker–Prager flow rule with tension cut-off in an unified manner. It included different ways of characterizing their sets of solutions, either with normal cone inclusions or as roots of complementarity functions. In Chapter 2, we recalled the Moreau–Jean timestepping algorithm, which allows to compute the dynamics of a mechanical system with bilateral constraints and unilateral contact with friction as a sequence of Discrete Coulomb Friction Problems (DCFP). We presented the Cadoux fixed-point algorithm, which yields both a criterion for the existence of solutions to a DCFP, and a practical way of solving this problem as a sequence of *convexified* DCFP (i.e., equivalently, as quadratic minimization problems under conical constraints). Chapter 3 then outlined standard algorithms for solving DCFP and convexified DCFP. Our first original contribution was presented in Chapter 4, and consisted in improving a numerical method for solving DCFP, targeted at ill-conditioned problems with a high number of contacts and for which reaching very high precision was not a requirement. To this aim, we proposed to equip the Gauss–Seidel algorithm with an hybrid local solver combining a novel Newton algorithm on the SOC Fischer–Burmeister function, and an enumerative analytical solver. We derived the expressions of the coefficients of the quartic polynomial that has

to be solved for both DCFP and convexified DCFP. We showed how this algorithm was relevant for applications to the direct and inverse simulation of hair and cloth, and provided benchmarks confirming its good practical performance. However, we noted that on smaller problems, a variant of the projected gradient descent algorithm which we coined ASPG also performed very well in practice. We also proposed a primal-dual algorithm "Dual AMA" requiring only multiplications by the stiffness matrix, but which failed to demonstrate consistent-enough convergence. These different solvers have been released as part of an open-source library developed alongside this thesis, `bogus` (Daviet 2013).

The second part of this dissertation built on our second contribution, presented in Chapter 6, a numerical strategy allowing to cast continuum granular flows problems into the framework of DCFP. Inspired by the work of Cadoux (2009), we showed how finding the solution to the non-associated Drucker–Prager flow rule over a continuous domain amounted once again to a sequence of minimization problems over a SOC in the space of symmetric tensor fields. However, we were only able to state a very weak existence criterion. We then demonstrated that by carefully choosing the basis for the discretization of the strain and stress fields, the discrete flow equations amounted to solving a DCFP (although with higher-dimensional constraints), and that we were able to leverage the solvers designed for standard discrete contact mechanics. We showed that this approach allowed us to capture qualitative features of dense granular flows (some of which being hard to achieve with standard methods, such as the vanishing pressure field in the wake of an obstacle), then extended our method to more general flows in Chapters 7 and 8. In particular, we proposed in Chapter 7 an approximation thanks to which our method proved to be efficient-enough for Computer Graphics applications, allowing us to significantly reduce visual artifacts w.r.t. a state-of-the-art method at a similar computational cost. However, we noted that the choice of suitable discretization spaces remained a notable difficulty. On the other hand, the soundness of our method allowed for easy generalization to more complex settings, be it anisotropic friction, two-way coupling with rigid bodies, or interactions with a surrounding fluid. This last point was the subject of our third main contribution, presented in Chapter 8. Starting from the framework of Jackson (2000), we proposed a two-phase, two-velocity model for the coupled simulation of a Drucker–Prager granular material with a Newtonian fluid. The main interesting feature of our approach is that, once again, we are able to cast the flow dynamics as a DCFP— though, in contrast to the single-phase case, one that includes linear equality constraints. While very limited at the moment, we presented a first few steps towards the qualitative validation of this model, including the capture of the different dynamical regimes governing the collapse of an immersed granular column reported in the literature.

9.2 Perspectives

Each step of our numerical method could probably be substantially improved: the choice of discretization spaces, the coupling between the transport and momentum conservation equations, the algorithm for solving the DCFP, *et cætera*. One of our initial goals that was finally excluded from the scope of this thesis was the devising of a massive parallelization strategy taking profit of the modern many-cores architectures. We have good hope that the proximal-based approaches such as ASPG or Dual AMA could scale quite well, even though sparse matrix–vector product is not the most parallelization-friendly routine. Indeed, our first tests in this domain indicated that for regular grids, the variance in the number of non-zeros blocks per row in the DCFP's matrices was quite small, and that storage schemes such as NVIDIA® cuSPARSE's HYB format[1] were particularly relevant in this case.

From a modeling point of view, we showed that it was possible to express the flow of a granular material, even inside a Newtonian fluid, in a compact, consistent manner and without any regularization as a sequence of numerical problems with a well-known structure. We demonstrated that that approach was able to capture some qualitative features of these coupled dynamics. Despite these first encouraging results, priority for future work should be put towards

[1] http://docs.nvidia.com/cuda/cusparse/#hybrid-format-hyb

additional validation w.r.t. experimental data or DEM simulations, and towards studying the influence of the several dimensionless numbers than appear in our diphasic model. Moreover, extending the numerical method of Chapter 8 to 3D simulations remains to be done before contemplating applications to Computer Graphics — which could include reproducing underwater sand avalanches or the dynamics of volcanic ash during an eruption.

Finally, while we were able to simulate the dynamics of fibrous materials consisting of a few thousand strands, larger systems still elude our DEM strategy. Moreover, air drag greatly influences the dynamics of hair, and modeling the coupled dynamics of air and hair would probably enhance realism. As coupling a fluid solver with hundreds of thousands of very fine DEM fibers would be extremely costly, continuum modeling of this interaction would seem more appropriate. An interesting research direction would thus be to attempt to extend our macroscopic numerical method to the simulation of hair and fur; however, several key ingredients of such media would still remain to be modeled. First, the bending elasticity of the fibers. One could imagine writing a macroscopic bending energy based on the total derivative the fibers' tangents field, or computing stresses on a discrete number of fiber samples in a MPM fashion. Stretching energy would probably have to be modeled in a similar manner, as the material cannot be macroscopically approximated as inextensible. Fibrous materials are also generally very anisotropic; the basic considerations that we laid down for coin-shaped grains in Chapter 7 would have to be generalized, and the distribution of fiber orientations should influence a much wider range of parameters, from the drag coefficient to the maximal volume fraction. Finally, our granular model cannot capture the natural entanglement of human hair, which induces a kind of *normal* (instead of *shearing*) friction. Applying Dahl's frictional model to this phenomenon would be an interesting starting point. Continuum modeling of slender, elastic fiber assemblies would thus require the introduction of several additional hypothesis and closures. Before attempting such an undertaking, it is therefore essential to get a good understanding of the simpler granular model.

Appendices

A Convex analysis

In this section we recall a few fundamental results from convex analysis, stemming for a large part from the early works of Jean-Jacques Moreau and R. Tyrrell Rockafellar. In the following, we mainly refer to the monograph "Fonctionnelles convexes", as it lists several results in very general settings — e.g., without assumption of finite dimension, even though here we will restrict ourselves to Banach or Hilbert spaces.

A.1 Operations on convex functions

A.1.1 Fundamental definitions

Let X be a reflexive Banach space on $\mathbb{R}$. Let $\bar{\mathbb{R}} := \mathbb{R} \cup \{-\infty, +\infty\}$ denote the extended-real set, and $f : X \to \bar{\mathbb{R}}$.

Definition A.1. *f is said to be convex if*

$$f(\alpha \boldsymbol{x} + (1-\alpha)\boldsymbol{z}) \leq \alpha f(\boldsymbol{x}) + (1-\alpha) \quad \forall \boldsymbol{x}, \boldsymbol{z}, \alpha \in X \times X \times]0,1[$$

where the addition is extended to $\bar{\mathbb{R}}$ with the commutative convention $(-\infty)+(+\infty) = +\infty$. f is said to be strictly convex if the above definition holds with strict inequality for all $\boldsymbol{x} \neq \boldsymbol{z}$.

Definition A.2 (Epigraph)**.** *The epigraph of f is the set* $\operatorname{epi} f := \{(\boldsymbol{x}, \alpha) \in X \times \mathbb{R}, f(\boldsymbol{x}) \leq \alpha\}$*. It holds that f is convex if and only if* $\operatorname{epi} f$ *is convex.*

Definition A.3 (Effective domain)**.** *The effective domain of f is the set* $\operatorname{dom} f := \{\boldsymbol{x} \in X, f(\boldsymbol{x}) < +\infty\}$*. f is said to be proper if* $\operatorname{dom} f \neq \emptyset$ *and* $\forall \boldsymbol{x} \in X, f(\boldsymbol{x}) > -\infty$*.*

Definition A.4 (Closed function)**.** *f is said to be closed if* $\operatorname{epi} f$ *is closed.*

Definition A.5. *[Lower semi-continuity] f is said to be lower semi-continuous at $\boldsymbol{x}_0 \in X$ if for all $\epsilon \in \mathbb{R}_+^*$, there exists a neighborhood V of $\boldsymbol{x}_0$ such that $\forall \boldsymbol{x} \in V$, $f(\boldsymbol{x}) \geq f(\boldsymbol{x}_0)) - \epsilon$.*

Property A.1. *The following propositions are equivalent (Moreau 1966–1967, paragraph 4.a):*

1. *f is closed*
2. *f is lower semi-continuous on X*
3. *$\forall \alpha \in \mathbb{R}$, the set $\{\boldsymbol{x} \in X, f(\boldsymbol{x}) \leq \alpha\}$ is closed.*

Property A.2. *Any proper closed convex function f is continuous on the interior of its effective domain (Moreau 1966–1967, paragraph 5.f).*

A.1.2 Subdifferential of a function

Let Y be the topological dual space of X (the space of continuous linear forms $\boldsymbol{y} : X \to \mathbb{R}$), with the bilinear form $\langle \cdot, \cdot \rangle : X \times Y \to \mathbb{R}$ such that Y can be identified to $\{\langle \cdot, \boldsymbol{y} \rangle, \boldsymbol{y} \in Y\}$ and X to $\{\langle \boldsymbol{x}, \cdot \rangle, \boldsymbol{x} \in X\}$.

Definition A.6 (Subdifferential). *If there exists $\boldsymbol{x}_0 \in X$ and $\boldsymbol{y} \in Y$ such that*

$$\forall \boldsymbol{x} \in X, \quad f(\boldsymbol{x}) \geq f(\boldsymbol{x}_0) + \langle \boldsymbol{x} - \boldsymbol{x}_0, \boldsymbol{y} \rangle$$

then f is said to be subdifferentiable at $\boldsymbol{x}_0$ and $\boldsymbol{y}$ is a subgradient of f at $\boldsymbol{x}_0$. We denote by $\partial f(\boldsymbol{x}_0)$ the subdifferential of f at $\boldsymbol{x}_0$, defined as the set of all subgradients at f at $\boldsymbol{x}_0$.

Remark A.1. *If f is proper, $\partial f(X \setminus \operatorname{dom} f) = \emptyset$.*

It follows from Definition A.6 that:

Property A.3 (Minimum of a convex function). *The global minimum of f is attained at $\boldsymbol{x}_0 \in X$ if and only if $\mathbf{0} \in \partial f(\boldsymbol{x}_0)$. Moreover, if f is strictly convex, this global minimum is attained at at most one point.*

Proof. $\mathbf{0} \in \partial f(\boldsymbol{x}_0) \iff f(\boldsymbol{x}) \geq f(\boldsymbol{x}_0)\ \forall \boldsymbol{x} \in X$, which is the definition of a global minimum. To prove the second assertion, suppose that f is strictly convex and attains its global minimum at two different points $\boldsymbol{x}_1 \neq \boldsymbol{x}_2$. Then $f(\frac{1}{2}(\boldsymbol{x}_1 + \boldsymbol{x}_2)) < \frac{1}{2}(f(\boldsymbol{x}_1) + f(\boldsymbol{x}_2)) = f(\boldsymbol{x}_1)$, which yields a contradiction. □

The following property suggests that the subdifferential can be seen as a generalization of the concept of the gradient of a convex function at points where it is not differentiable.

Property A.4. *Let f be a proper convex function on X, continuous at $\boldsymbol{x}_0 \in \operatorname{dom}(f)$. The following statements are equivalent (Moreau 1966–1967, paragraphs 10.f and 10.g)*

- *f is Gâteaux-differentiable at $\boldsymbol{x}_0$ with gradient $(\nabla f)(\boldsymbol{x}_0) \in Y$, i.e.,*

$$\lim_{\alpha \to 0_+} \frac{f(\boldsymbol{x}_0 + \alpha \boldsymbol{x}) - f(\boldsymbol{x}_0)}{\alpha} = \langle \boldsymbol{x}, (\nabla f)(\boldsymbol{x}_0) \rangle \quad \forall \boldsymbol{x} \in X;$$

- $\partial f(\boldsymbol{x}_0) = \{(\nabla f)(\boldsymbol{x}_0)\}$.

Remark A.2. *If f is Fréchet-differentiable, then it is continuous and Gâteaux-differentiable as well, and the gradients from both definitions coincide. The converse is not true; intuitively Fréchet ensures that the differential is well-defined along all continuous paths converging to $\boldsymbol{x}_0$, while Gâteaux considers only straight lines.*

Sum of differentials It always hold that $\partial f + \partial g \subset \partial(f + g)$, but some regularity (*a.k.a.* qualification) conditions are required for the equality to be achieved. Moreau (1966–1967, Proposition 10.7) states the following theorem:

Theorem A.1 (Subdifferential of a sum). *For f and g convex $X \to \bar{\mathbb{R}}$, if there exists $\boldsymbol{x}_0 \in \operatorname{dom} f \cap \operatorname{dom} g$ such that either f or g is continuous at $\boldsymbol{x}_0$, then $\partial(f + g) = \partial f + \partial g$ on X.*

As a proper convex function is continuous on the interior of its effective domain (Property A.2), it follows:

Corollary A.1. *Let f and g be convex functions $X \to \bar{\mathbb{R}}$, with f closed and proper. If $(\operatorname{int} \operatorname{dom} f) \cap \operatorname{dom} g \neq \emptyset$, then $\partial(f + g) = \partial f + \partial g$ on X.*

Special cases The following property will prove useful in continuum mechanics.

Property A.5 (Subdifferential in function spaces). *Let Ω be a subset of $\mathbb{R}^d$, $d \in \mathbb{N}$, and let $L_2(\Omega)$ denote the space of square-integrable functions on Ω. Let $\langle \cdot, \cdot \rangle$ be a scalar product on $\mathbb{R}^n$, $n \in \mathbb{N}$; we equip the space $V := L_2(\Omega)^n$ with the scalar product $\langle \cdot, \cdot \rangle_V$,*

$$\langle \mathbf{u}, \mathbf{v} \rangle_V := \int_\Omega \langle \mathbf{u}(\boldsymbol{x}), \mathbf{v}(\boldsymbol{x}) \rangle \, d\boldsymbol{x} \quad \forall (\mathbf{u}, \mathbf{v}) \in V^2.$$

Finally, consider a function $g : V \to \bar{\mathbb{R}}$, $\mathbf{u} \mapsto \int_\Omega f(\mathbf{u}(\boldsymbol{x}))d\boldsymbol{x}$, *where* $f : \mathbb{R}^n \to \mathbb{R}$. *The subdifferential of* g *is given by, for any* $\mathbf{u} \in V$,

$$\partial g(\mathbf{u}) = \{\mathbf{v} \in V, \mathbf{v}(\boldsymbol{x}) \in \partial f(\mathbf{u}(\boldsymbol{x})) \text{ a.e. on } \Omega\}.$$

Proof. V is an Hilbert space, thus $\partial g \subset V$. Let $\mathbf{u}, \mathbf{v} \in V^2$, $\Omega_\mathbf{v}^+ := \{\boldsymbol{x} \in \Omega, \mathbf{v}(\boldsymbol{x}) \in \partial f(\mathbf{u}(\boldsymbol{x}))\}$ and $\Omega_\mathbf{v}^- := \Omega - \Omega_\mathbf{v}^+$. By definition of the subdifferential, $\boldsymbol{x} \in \Omega_\mathbf{v}^+ \iff \forall \boldsymbol{z} \in \mathbb{R}^d$, $f(\boldsymbol{z}) - f(\mathbf{u}(\boldsymbol{x})) \geq \langle \mathbf{v}(\boldsymbol{x}), \boldsymbol{z} - \mathbf{u}(\boldsymbol{x})\rangle$.

For any $\mathbf{w} \in V$, we thus have

$$\begin{aligned} g(\mathbf{w}) - g(\mathbf{u}) &= \int_{\Omega_\mathbf{v}^+} f(\mathbf{w}(\boldsymbol{x})) - f(\mathbf{u}(\boldsymbol{x})) + \int_{\Omega_\mathbf{v}^-} f(\mathbf{w}(\boldsymbol{x})) - f(\mathbf{u}(\boldsymbol{x})) \\ &\geq \int_{\Omega_\mathbf{v}^+} \langle \mathbf{v}(\boldsymbol{x}), \mathbf{w}(\boldsymbol{x}) - \mathbf{u}(\boldsymbol{x})\rangle + \int_{\Omega_\mathbf{v}^-} f(\mathbf{w}(\boldsymbol{x})) - f(\mathbf{u}(\boldsymbol{x})). \end{aligned}$$

As f takes its values in $\mathbb{R}$, $\Omega_\mathbf{v}^-$ being of measure zero implies that $g(\mathbf{w}) - g(\mathbf{x}) \geq \langle \mathbf{v}, \mathbf{w} - \mathbf{u}\rangle_V$, and thus $\mathbf{v} \in \partial g(\mathbf{u})$. Reciprocally, suppose that $\Omega_\mathbf{v}^-$ is not of measure zero. By definition of this set, for all $\boldsymbol{x} \in \Omega_\mathbf{v}^-$ there exists $\boldsymbol{z}_\boldsymbol{x} \in \mathbb{R}^d$ such that $f(\boldsymbol{z}_\boldsymbol{x}) - f(\mathbf{u}(\boldsymbol{x})) < \langle \mathbf{v}(\boldsymbol{x}), \boldsymbol{z}_\boldsymbol{x} - \mathbf{u}(\boldsymbol{x})\rangle$. Let us define $\mathbf{w} \in V$ as

$$\mathbf{w}(\boldsymbol{x}) := \begin{cases} \boldsymbol{z}_\boldsymbol{x} & \text{on a relatively compact subset of } \Omega_\mathbf{v}^- \text{ with strictly positive measure} \\ \mathbf{u}(\boldsymbol{x}) & \text{elsewhere.} \end{cases}$$

Then $g(\mathbf{w}) - g(\mathbf{u}) = \int_{\Omega_\mathbf{v}^-} \boldsymbol{z}_\boldsymbol{x} - f(\mathbf{u}(\boldsymbol{x})) < \int_{\Omega_\mathbf{v}^-} \langle \mathbf{v}(\boldsymbol{x}), \boldsymbol{z}_\boldsymbol{x} - \mathbf{u}(\boldsymbol{x})\rangle = \langle \mathbf{v}, \mathbf{w} - \mathbf{u}\rangle_V$, i.e., $\mathbf{v} \notin \partial g(\mathbf{u})$. □

Finally, we state a property of subdifferentials that will not be directly useful in this dissertation, but allows the easy extension of results derived for a scalar constraint to larger dimensional ones. The proof is given for a finite-dimensional space in (Hiriart-Urruty and Lemaréchal 1993, VI, Corollary 4.3.2), and extended to infinite sequences in (Hiriart-Urruty and Lemaréchal 1993, VI, Theorem 4.4.2).

Property A.6 (Subdifferential of a maximum)**.** *Let* f_i, $i = 1 \dots n$ *be a finite number of convex functions on* X. *Then the subdifferential of their point-wise maximum a point* $\boldsymbol{x}_0$ *is the convex hull of the subdifferentials of the* f_i *that are active at* $\boldsymbol{x}_0$, *i.e.,*

$$\partial(\max(f_i))(\boldsymbol{x}_0) = \text{Conv} \bigcup_{j \in J} \partial f_j \text{ with } J := \{j, f_j(\boldsymbol{x}_0) = \max_i f_i(\boldsymbol{x}_0))\}.$$

A.1.3 Convex conjugate

Definition A.7 (Convex conjugate)**.** *The convex conjugate (or Fenchel–Legendre transform) of* f *is the application*

$$\begin{aligned} f^\star : Y &\to \bar{\mathbb{R}} \\ \boldsymbol{y} &\mapsto \sup_{\boldsymbol{x} \in X} (\langle \boldsymbol{x}, \boldsymbol{y}\rangle - f(\boldsymbol{x})). \end{aligned}$$

Moreau (1966–1967, paragraphs 6.b and 6.d) gives two fundamental properties:

Property A.7. *The convex conjugate of* f *is always convex and closed. If* f *is proper, convex and closed, then it is equal to its biconjugate, i.e.,* $f = f^{\star\star}$.

Theorem A.2 (Subnormality)**.** *It always hold that* $f(\boldsymbol{x}) + f^\star(\boldsymbol{y}) \geq \langle \boldsymbol{x}, \boldsymbol{y}\rangle$ *(Fenchel-Young inequality). Moreover, if* f *is proper, closed and convex, then*

$$\boldsymbol{y} \in \partial f(\boldsymbol{x}) \iff f(\boldsymbol{x}) + f^\star(\boldsymbol{y}) = \langle \boldsymbol{x}, \boldsymbol{y}\rangle \iff \boldsymbol{x} \in \partial f^\star(\boldsymbol{y}).$$

Proof. From the definition of the convex conjugate, $\forall \boldsymbol{x}, \boldsymbol{y} \in X \times Y$,

$$\begin{aligned} f(\boldsymbol{x}) + f^\star(\boldsymbol{y}) =& f(\boldsymbol{x}) + \sup_{\boldsymbol{z} \in X} (\langle \boldsymbol{z}, \boldsymbol{y} \rangle - f(\boldsymbol{z})) \\ \geq& f(\boldsymbol{x}) + (\langle \boldsymbol{x}, \boldsymbol{y} \rangle - f(\boldsymbol{x})) = \langle \boldsymbol{x}, \boldsymbol{y} \rangle \end{aligned}$$

Then,

$$\begin{aligned} f(\boldsymbol{x}) + f^\star(\boldsymbol{y}) = \langle \boldsymbol{x}, \boldsymbol{y} \rangle &\iff f^\star(\boldsymbol{y}) = \langle \boldsymbol{x}, \boldsymbol{y} \rangle - f(\boldsymbol{x}) \\ &\iff \langle \boldsymbol{z}, \boldsymbol{y} \rangle - f(\boldsymbol{z}) \leq \langle \boldsymbol{x}, \boldsymbol{y} \rangle - f(\boldsymbol{x}) && \forall \boldsymbol{z} \in X \\ &\iff f(\boldsymbol{x}) + \langle \boldsymbol{z} - \boldsymbol{x}, \boldsymbol{y} \rangle \leq f(\boldsymbol{z}) && \forall \boldsymbol{z} \in X \\ &\iff \boldsymbol{y} \in \partial f(\boldsymbol{x}) \end{aligned}$$

The second part of the equivalence stems from $f^{\star\star} = f$ (Property A.7). □

We transcribe below a slightly weakened version of the Fenchel duality theorem as given in (Borwein and Luke 2011, theorem 1.2),

Theorem A.3 (Fenchel duality)**.** *Let X and E be Banach spaces, with $E^\star$ the dual of E. Let L be a bounded linear map.*

Then the extrema of the optimization problems (A.1) and (A.2),

$$p = \inf_{\boldsymbol{x} \in X} f(\boldsymbol{x}) + g(L\boldsymbol{x}) \tag{A.1}$$

$$d = \sup_{\boldsymbol{z} \in E^\star} -f^\star(L^\intercal \boldsymbol{z}) - g^\star(-\boldsymbol{z}) \tag{A.2}$$

satisfy weak duality, i.e., $p \geq d$.

Moreover if f and g are convex and if there exists $\boldsymbol{x} \in L \operatorname{dom} f$ such that $\boldsymbol{g}$ is finite and continuous at $\boldsymbol{x}_0$, strong duality holds: $p = d$, and if they are finite, then the dual problem (A.2) attains its maximum.

Corollary A.2. *Extension to affine maps; L in the theorem above is replaced by $A : \boldsymbol{x} \mapsto L\boldsymbol{x} + \boldsymbol{e}$, and the optimization problems (A.1) and (A.2) with*

$$\begin{aligned} p &= \inf_{\boldsymbol{x} \in X} f(\boldsymbol{x}) + (g \circ A)(\boldsymbol{x}) \\ d &= \sup_{\boldsymbol{z} \in E^\star} -f^\star(L^\intercal \boldsymbol{z}) - g^\star(-\boldsymbol{z}) - \langle \boldsymbol{e}, \boldsymbol{z} \rangle \end{aligned}$$

Weak duality always hold, and strong duality is achieved when f and g are convex and there exists $\boldsymbol{x} \in A \operatorname{dom} f$ such that $\boldsymbol{g}$ is finite and continuous at $\boldsymbol{x}_0$.

Proof. By application of the theorem to $g_e : \boldsymbol{z} \mapsto g(\boldsymbol{z} + \boldsymbol{e})$; indeed $g_e(L\boldsymbol{x}) = g \circ A(\boldsymbol{x})$, and $({}^\star g_e)(-\boldsymbol{z}) = ({}^\star g)(-\boldsymbol{z}) + \langle \boldsymbol{e}, \boldsymbol{z} \rangle$. □

A.2 Normal and convex cones

A.2.1 Normal cone

Definition A.8 (Characteristic function)**.** *The characteristic function $\mathscr{I}_C$ of a set $C \subset X$ is defined as vanishing on C and taking infinite values outside of C,*

$$\mathscr{I}_C := \begin{cases} 0 & \text{if } \boldsymbol{x} \in C \\ +\infty & \text{if } \boldsymbol{x} \notin C. \end{cases}$$

From $\operatorname{epi} \mathscr{I}_C = C \times \mathbb{R}^+$ and $\operatorname{dom} \mathscr{I}_C = C$ it follows:

- $\mathscr{I}_C$ is convex if and only if C is convex.

- $\mathscr{I}_C$ is not empty if and only if C is proper.
- $\mathscr{I}_C$ is closed if and only if C is closed.

Definition A.9 (Normal cone)**.** *Let C be a convex subset of X. The normal cone to C at $\boldsymbol{x}_0 \in X$ is defined as*

$$\mathscr{N}_C(\boldsymbol{x}_0) = \begin{cases} \{\boldsymbol{y} \in Y, \langle \boldsymbol{x} - \boldsymbol{x}_0, \boldsymbol{y} \rangle \leq 0 \quad \forall \boldsymbol{x} \in C\} & \text{if } \boldsymbol{x}_0 \in C \\ \emptyset & \text{if } \boldsymbol{x}_0 \notin C \end{cases}$$

As it can easily be seen that for any point $\boldsymbol{x}_0$ in C, $\mathbf{0} \in \mathscr{N}_C(\boldsymbol{x}_0)$, we have the equivalence:

Property A.8. $\mathbf{0} \in \mathscr{N}_C(\boldsymbol{x}) \iff \boldsymbol{x} \in C$

Property A.9. *If C is not empty, there holds $\mathscr{N}_C = \partial \mathscr{I}_C$.*

Proof. As $\mathscr{I}_C$ is proper, from Remark A.1 we have $\boldsymbol{x}_0 \notin C \iff \partial \mathscr{I}_C(\boldsymbol{x}_0) = \emptyset = \mathscr{N}_C(\boldsymbol{x}_0)$. For $\boldsymbol{x}_0 \in C$, Definition A.6 reads

$$\begin{aligned} \boldsymbol{y} \in \partial \mathscr{I}_C(\boldsymbol{x}_0) &\iff \mathscr{I}_C(\boldsymbol{x}) \geq \mathscr{I}_C(\boldsymbol{x}_0) + \langle \boldsymbol{x} - \boldsymbol{x}_0, \boldsymbol{y} \rangle && \forall \boldsymbol{x} \in X \\ &\iff \mathscr{I}_C(\boldsymbol{x}) \geq \langle \boldsymbol{x} - \boldsymbol{x}_0, \boldsymbol{y} \rangle && \forall \boldsymbol{x} \in X \\ &\iff 0 \geq \langle \boldsymbol{x} - \boldsymbol{x}_0, \boldsymbol{y} \rangle && \forall \boldsymbol{x} \in C \\ &\iff \boldsymbol{y} \in \mathscr{N}_C(\boldsymbol{x}_0) \end{aligned}$$

□

We deduce immediately that the expression of the normal cone at an interior point.

Property A.10. $\boldsymbol{x} \in \operatorname{int} C \implies \mathscr{N}_C(\boldsymbol{x}) = \{\mathbf{0}\}$.

Proof. For $\boldsymbol{x} \in \operatorname{int} C$, there exists a closed ball with strictly positive radius centered at $\boldsymbol{x}$ on which $\mathscr{I}_C$ is constant. Therefore $\mathscr{I}_C$ is continuous and Gâteaux-differentiable at $\boldsymbol{x}$, with $(\nabla \mathscr{I}_C)(\boldsymbol{x}_0) = \mathbf{0}$. Properties A.4 and A.9 conclude the proof. □

A.2.2 Operations on normal cones

Property A.11 (Normal cone to an intersection)**.** *For any two subsets C_1 and C_2 of X, $\mathscr{N}_{C_1 \cap C_2} \supset \mathscr{N}_{C_1} + \mathscr{N}_{C_2}$.*

Moreover, if C_1 and C_2 are convex, with C_1 closed and $(\operatorname{int} C_1) \cap C_2 \neq \emptyset$, then $\mathscr{N}_{C_1 \cap C_2} = \mathscr{N}_{C_1} + \mathscr{N}_{C_2}$.

Proof. If C_1 and C_2 are empty, the inclusion is trivial. Otherwise, we have $\mathscr{I}_{C_1 \cap C_2} = \mathscr{I}_{C_1} + \mathscr{I}_{C_2}$, therefore with Property A.9,

$$\mathscr{N}_{C_1 \cap C_2} = \partial\left(\mathscr{I}_{C_1} + \mathscr{I}_{C_2}\right) \supset \partial \mathscr{I}_{C_1} + \partial \mathscr{I}_{C_2} = \mathscr{N}_{C_1} + \mathscr{N}_{C_2}.$$

Now, if C_1 and C_2 are convex with C_1 closed and $(\operatorname{int} C_1) \cap C_2 \neq 0$, the regularity conditions of Corollary A.1 on the subdifferential of a sum are satisfied, and the equality holds. □

Corollaries of the following property will often prove useful for the practical computation of normal cones.

Property A.12 (Subdifferential of the precomposition by an affine map)**.** *Let X and E be Hilbert spaces, and A an affine map from X to E, i.e., $A(\boldsymbol{x}) = L\boldsymbol{x} + \boldsymbol{e}$ with L linear and $\boldsymbol{e} \in E$. We note $L^\intercal$ the linear adjoint of L. Let f be a convex function on E. There holds, for all $\boldsymbol{x} \in X$,*

$$\partial(f \circ A)(\boldsymbol{x}) \supset L^\intercal \partial f(A(\boldsymbol{x})).$$

Moreover, if f is proper and closed, and $\exists \boldsymbol{x}_0 \in \operatorname{dom}(f \circ A)$ such that f is continuous at $A(\boldsymbol{x}_0)$, then for all $\boldsymbol{x} \in X$,

$$\partial(f \circ A)(\boldsymbol{x}) = L^\intercal \partial f(A(\boldsymbol{x})).$$

Proof. First, let $\boldsymbol{y} \in L^\intercal \partial f(A(\boldsymbol{x}_0))$. There exists $\boldsymbol{g} \in E$ such that $\boldsymbol{y} = L^\intercal \boldsymbol{g}$ and

$$f(\boldsymbol{z}) \geq f(A(\boldsymbol{x}_0)) + \langle \boldsymbol{z} - A(\boldsymbol{x}_0), \boldsymbol{g} \rangle \qquad \forall \boldsymbol{z} \in E.$$

In particular,

$$f(A(\boldsymbol{x})) \geq f(A(\boldsymbol{x}_0)) + \langle A(\boldsymbol{x}) - A(\boldsymbol{x}_0), \boldsymbol{g} \rangle \qquad \forall \boldsymbol{x} \in X.$$

Since $\langle A(\boldsymbol{x}) - A(\boldsymbol{x}_0), \boldsymbol{g} \rangle = \langle L\boldsymbol{x} - L\boldsymbol{x}_0, \boldsymbol{g} \rangle = \langle \boldsymbol{x} - \boldsymbol{x}_0, L^\intercal \boldsymbol{g} \rangle$,

$$f(A(\boldsymbol{x})) \geq f(A(\boldsymbol{x}_0)) + \langle \boldsymbol{x} - \boldsymbol{x}_0, \boldsymbol{y} \rangle \qquad \forall \boldsymbol{x} \in X,$$

which means that $\boldsymbol{y} \in \partial(f \circ A)(\boldsymbol{x}_0)$.

Now, let us prove the reverse inclusion under our supplemental regularity conditions. Let $\boldsymbol{y} \in \partial(f \circ A)(\boldsymbol{x}_0)$, and let $\boldsymbol{x}^* \in \mathrm{dom}(f \circ A)$, which is not empty under our hypothesis. Then, $\forall \boldsymbol{x} \in \mathrm{Ker}\, L$,

$$\begin{aligned} f(A(\boldsymbol{x}^* + \boldsymbol{x})) &\geq f(A(\boldsymbol{x}_0)) + \langle \boldsymbol{x}^* + \boldsymbol{x} - \boldsymbol{x}_0, \boldsymbol{y} \rangle \\ -\langle \boldsymbol{x}, \boldsymbol{y} \rangle &\geq f(A(\boldsymbol{x}_0)) - f(A(\boldsymbol{x}^*)) + \langle \boldsymbol{x}^* - \boldsymbol{x}_0, \boldsymbol{y} \rangle . \end{aligned}$$

As the right-hand side is finite, for the inequality to hold for all $\boldsymbol{x} \in \mathrm{Ker}\, L$, we must have $\boldsymbol{y} \in (\mathrm{Ker}\, L)^\perp = \mathrm{Im}\, L^\intercal$, i.e., $\exists \boldsymbol{g} \in E$, $\boldsymbol{y} = L^\intercal \boldsymbol{g}$. Going back to the definition of the subdifferential, we must also have

$$\begin{aligned} f(A(\boldsymbol{x})) &\geq f(A(\boldsymbol{x}_0)) + \langle \boldsymbol{x} - \boldsymbol{x}_0, L^\intercal \boldsymbol{g} \rangle && \forall \boldsymbol{x} \in X \\ f(A(\boldsymbol{x})) &\geq f(A(\boldsymbol{x}_0)) + \langle A(\boldsymbol{x}) - A(\boldsymbol{x}_0), \boldsymbol{g} \rangle && \forall \boldsymbol{x} \in X \\ f(\boldsymbol{z}) &\geq f(A(\boldsymbol{x}_0)) + \langle \boldsymbol{z} - A(\boldsymbol{x}_0), \boldsymbol{g} \rangle && \forall \boldsymbol{z} \in \mathrm{Im} A \\ (f + \mathscr{I}_{\mathrm{Im}A})(\boldsymbol{z}) &\geq (f + \mathscr{I}_{\mathrm{Im}A})(A(\boldsymbol{x}_0)) + \langle \boldsymbol{z} - A(\boldsymbol{x}_0), \boldsymbol{g} \rangle && \forall \boldsymbol{z} \in E, \end{aligned}$$

which means $\boldsymbol{g} \in \partial(f + \mathscr{I}_{\mathrm{Im}A})(A(\boldsymbol{x}_0))$. Under our regularity conditions, we can apply Theorem A.1 and get equivalently $\boldsymbol{g} \in \partial f(A(\boldsymbol{x}_0)) + \partial \mathscr{I}_{\mathrm{Im}A}(A(\boldsymbol{x}_0))$. As $\partial \mathscr{I}_{\mathrm{Im}A}(A(\boldsymbol{x}_0)) = \mathscr{N}_{\mathrm{Im}A}(A(\boldsymbol{x}_0)) = \mathrm{Ker}\, L^\intercal$, we get

$$\boldsymbol{y} \in L^\intercal(\partial f(A(\boldsymbol{x}_0)) + \mathrm{Ker}\, L^\intercal) = L^\intercal \partial f(A(\boldsymbol{x}_0)).$$

□

We can get the following corollary by choosing $f = \mathscr{I}_C$ in the above property:

Corollary A.3. *Let C be a closed convex subset of E such that $\mathrm{int}\, C \cap \mathrm{Im} A \neq 0$. Then the normal cone to $\Gamma := \{\boldsymbol{x} \in X,\ A(\boldsymbol{x}) \in C\}$ is*

$$\mathscr{N}_\Gamma(\boldsymbol{x}) = L^\intercal \mathscr{N}_C(A(\boldsymbol{x}))$$

The extension to linear constraints stated below will also prove useful for mechanical systems.

Corollary A.4. *Let $A : X \to E$ be an affine map defined as in Property A.12, i.e., $A : \boldsymbol{x} \mapsto L\boldsymbol{x} + \boldsymbol{e}$, $B : X \to F$ a linear operator from X to F Hilbert space, and for any $\boldsymbol{f} \in F$, $V(\boldsymbol{f}) := \{\boldsymbol{x} \in X, B\boldsymbol{x} = \boldsymbol{f}\}$. Let C be a closed convex subset of E, then the normal cone to $\Gamma(\boldsymbol{f}) := \{\boldsymbol{x} \in V(\boldsymbol{f}),\ A(\boldsymbol{x}) \in C\}$ satisfies*

$$\mathscr{N}_{\Gamma(\boldsymbol{f})}(\boldsymbol{x}) \supset L^\intercal \mathscr{N}_C(A(\boldsymbol{x})) + B^\intercal \mathscr{N}_{\{\boldsymbol{0}_F\}}(B\boldsymbol{x} - \boldsymbol{f}),$$

with equality under the regularity condition

$$\mathrm{int}\, C \cap A(V(\boldsymbol{f})) \neq \emptyset. \tag{A.3}$$

Proof. The inclusion being trivial, we focus on the equality. The regularity condition (A.3) requires that $V(\boldsymbol{f})$ is not empty; let $\boldsymbol{v}_f \in V(\boldsymbol{f})$, then $V(\boldsymbol{f}) = V(\boldsymbol{0}_F) + \boldsymbol{v}_f$.

$V(\boldsymbol{0}_F) = \mathrm{Ker}\, B$ is a linear subspace of X and is therefore a Hilbert space. As the regularity condition (A.3) can be written

$$\mathrm{int}\, C \cap \left(LV(\boldsymbol{0}_F) + \boldsymbol{e} + L\boldsymbol{v}_f\right) \neq \emptyset$$

we can apply Corollary A.3 to the linear map $A_0 : V(\mathbf{0}_F) \to E$, $\boldsymbol{x} \mapsto L\boldsymbol{x} + (\boldsymbol{e} + L\boldsymbol{v}_f)$, and get that the normal cone to $\Gamma_0(\boldsymbol{f}) \subset V(\mathbf{0}_F) := \{\boldsymbol{x} \in V(\mathbf{0}_F),\ A_0(\boldsymbol{x}) \in C\}$ is

$$\mathcal{N}_{\Gamma_0(f)}(\boldsymbol{x}) = L^\intercal \mathcal{N}_C(A_0(\boldsymbol{x})).$$

Now, suppose $\boldsymbol{y} \in \mathcal{N}_{\Gamma(f)}(\boldsymbol{x})$. Necessarily $\boldsymbol{x} \in \Gamma(\boldsymbol{f})$, and thus $\boldsymbol{x} - \boldsymbol{v}_f \in V(\mathbf{0}_F)$. Moreover, we can decompose $\boldsymbol{y}$ as $\boldsymbol{y} = \boldsymbol{y}_{|V(\mathbf{0}_F)} + \boldsymbol{y}_\perp$, with $\boldsymbol{y}_{|V(\mathbf{0}_F)} \in V(\mathbf{0}_F)$ and $\boldsymbol{y}_\perp \in \operatorname{Im} B^\intercal$. Therefore,

$$\begin{aligned}
&\forall \boldsymbol{z} \in \Gamma(\boldsymbol{f}), && \langle \boldsymbol{y}, \boldsymbol{z} - \boldsymbol{x} \rangle \leq 0 \\
&\forall \boldsymbol{z} \in \Gamma_0(\boldsymbol{f}), && \langle \boldsymbol{y}, \boldsymbol{z} - (\boldsymbol{x} - \boldsymbol{v}_f) \rangle \leq 0 \\
&\forall \boldsymbol{z} \in \Gamma_0(\boldsymbol{f}), && \langle \boldsymbol{y}_{|V(\mathbf{0}_F)}, \boldsymbol{z} - (\boldsymbol{x} - \boldsymbol{v}_f) \rangle \leq 0,
\end{aligned}$$

and necessarily $\boldsymbol{y} \in \mathcal{N}_{\Gamma_0}(\boldsymbol{x} - \boldsymbol{v}_f) + B^\intercal \mathcal{N}_{\{\mathbf{0}_F\}}(B\boldsymbol{x} - \boldsymbol{f})$. As the reverse inclusion always holds from Property A.11 on the normal cone to an intersection and Corollary A.3 applied to the linear constraint, we get

$$\begin{aligned}
\mathcal{N}_{\Gamma(f)}(\boldsymbol{x}) &= L^\intercal \mathcal{N}_C(A_0(\boldsymbol{x} - \boldsymbol{v}_f)) + B^\intercal \mathcal{N}_{\{\mathbf{0}_F\}}(B\boldsymbol{x} - \boldsymbol{f}) \\
&= L^\intercal \mathcal{N}_C(A(\boldsymbol{x})) + B^\intercal \mathcal{N}_{\{\mathbf{0}_F\}}(B\boldsymbol{x} - \boldsymbol{f}).
\end{aligned}$$

□

A.2.3 Convex cones

The name *normal cone* is coined from the fact that its belong to the class of *convex cones*, which possesses interesting properties w.r.t. the convex conjugate.

Definition A.10 (Convex cone)**.** *A subset $K \subset X$ is a convex cone if*

$$\boldsymbol{x}, \boldsymbol{y} \in K^2 \implies (\alpha \boldsymbol{x} + \beta \boldsymbol{y}) \in K, \quad \forall \alpha, \beta \in \mathbb{R}_+^2.$$

Property A.13. *There holds:*

1. *For any $\boldsymbol{x} \in C \subset X$, $\mathcal{N}_C(\boldsymbol{x})$ is a convex cone.*
2. *If K is a convex cone, then $\mathbf{0} \in K \iff K \neq \emptyset$.*

Definition A.11 (Dual and polar cones)**.** *The polar cone of $C \subset X$ is the set*

$$C^\circ := \{\boldsymbol{y} \in Y,\ \langle \boldsymbol{x}, \boldsymbol{y} \rangle \leq 0 \quad \forall \boldsymbol{x} \in C\}.$$

The dual cone of C is then defined as the negative of the polar cone, $C^\star := -C^\circ$.

Property A.14 (Normal cone to a convex cone)**.** *If K is a convex cone, then*

$$\mathcal{N}_K(\boldsymbol{x}) = \begin{cases} \emptyset & \text{if } \boldsymbol{x} \notin K \\ K^\circ \cap \{\boldsymbol{x}\}^\perp & \text{if } \boldsymbol{x} \in K. \end{cases}$$

Proof. The first case holds for any normal cone (Property A.8). Let us prove the second one.

First, we choose $\boldsymbol{x} \in K$ and $\boldsymbol{y} \in \mathcal{N}_K(\boldsymbol{x})$, i.e., $\forall \boldsymbol{x} \in K, \langle \boldsymbol{z} - \boldsymbol{x}, \boldsymbol{y} \rangle \leq 0$. Since $\mathbf{0} \in K$, $\langle \mathbf{0} - \boldsymbol{x}, \boldsymbol{y} \rangle \leq 0$, and since $2\boldsymbol{x} \in K$ as well, $\langle \boldsymbol{x}, \boldsymbol{y} \rangle \leq 0$, therefore $\langle \boldsymbol{x}, \boldsymbol{y} \rangle = 0$, i.e., $\boldsymbol{y} \in \{\boldsymbol{x}\}^\perp$.

Moreover, $\forall \boldsymbol{z} \in K, (\boldsymbol{x} + \boldsymbol{z}) \in K$, meaning that $\forall \boldsymbol{z} \in K, \langle \boldsymbol{z}, \boldsymbol{y} \rangle \leq 0$, i.e., $\boldsymbol{y} \in K^\circ$. Necessarily, we therefore have $\mathcal{N}_K(\boldsymbol{x}) \subset K^\circ \cap \{\boldsymbol{x}\}^\perp$. Let us now prove that the inclusion is not strict. Let $\boldsymbol{y} \in K^\circ \cap \{\boldsymbol{x}\}^\perp$, and $\boldsymbol{z} \in K$. Then $\langle \boldsymbol{z} - \boldsymbol{x}, \boldsymbol{y} \rangle = \langle \boldsymbol{z}, \boldsymbol{y} \rangle - \langle \boldsymbol{x}, \boldsymbol{y} \rangle \leq 0$ from the definitions of the polar cone and of the orthogonal subspace. Since this is true $\forall \boldsymbol{z} \in K$, $\boldsymbol{y} \in \mathcal{N}_K(\boldsymbol{x})$. Therefore, $\forall \boldsymbol{x} \in K$, $\mathcal{N}_K(\boldsymbol{x}) = K^\circ \cap \{\boldsymbol{x}\}^\perp$. □

Property A.15. *For K a non-empty convex cone, $(\mathscr{I}_K)^\star = \mathscr{I}_{K^\circ}$*

Proof. Following from the definition of the convex conjugate,

$$\begin{aligned}\mathscr{I}_K^\star(\boldsymbol{y}) &= \sup_{\boldsymbol{x}\in X}(\langle \boldsymbol{x},\boldsymbol{y}\rangle - \mathscr{I}_K(\boldsymbol{x})) \\ &= \sup_{\boldsymbol{x}\in K}(\langle \boldsymbol{x},\boldsymbol{y}\rangle)\end{aligned}$$

As K is a cone, $(0\boldsymbol{x}) \in K$ and therefore $\mathscr{I}_K^\star(\boldsymbol{y}) \geq 0$. Moreover if $\exists \boldsymbol{x}_0 \in K, \langle \boldsymbol{x}_0, \boldsymbol{y}\rangle > 0$, then $\forall \beta \in \mathbb{R}_+$, we can choose $\alpha = \frac{\beta}{\langle \boldsymbol{x}_0,\boldsymbol{y}\rangle}$ so that $(\alpha \boldsymbol{x}_0) \in K$ and $\langle \alpha\boldsymbol{x}_0, \boldsymbol{y}\rangle \geq \beta$, i.e., $\mathscr{I}_K^\star(\boldsymbol{y}) = +\infty$.

This means

$$\mathscr{I}_K^\star(\boldsymbol{y}) = \begin{cases} +\infty & \text{if } \exists \boldsymbol{x} \in K,\ \langle \boldsymbol{x},\boldsymbol{y}\rangle > 0 \\ 0 & \text{otherwise}\end{cases} = \begin{cases} +\infty & \text{if } \boldsymbol{x} \notin K^\circ \\ 0 & \text{otherwise}\end{cases} = \mathscr{I}_{K^\circ}.$$

□

Corollary A.5. *A non-empty closed convex cone K is the dual (resp., polar) cone of its dual (resp., polar) cone, i.e., $K^{\star\star} = K$ and $K^{\circ\circ} = K$.*

Proof.

$$\begin{aligned}
\boldsymbol{x} \in K^{\circ\circ} &\iff \boldsymbol{x} \in \mathscr{N}_{K^\circ}(\boldsymbol{0}) && \text{using Property A.14} \\
&\iff \boldsymbol{x} \in \partial \mathscr{I}_{K^\circ}(\boldsymbol{0}) && \text{using Property A.9} \\
&\iff \boldsymbol{x} \in \partial \mathscr{I}_K^\star(\boldsymbol{0}) && \text{using Property A.15} \\
&\iff \boldsymbol{0} \in \partial \mathscr{I}_K^{\star\star}(\boldsymbol{x}) && \text{using Theorem A.2} \\
&\iff \boldsymbol{0} \in \partial \mathscr{I}_K(\boldsymbol{x}) && \text{using Property A.7} \\
&\iff \boldsymbol{0} \in \mathscr{N}_K(\boldsymbol{x}) && \text{using Property A.9} \\
&\iff \boldsymbol{x} \in K && \text{using Property A.8.}
\end{aligned}$$

□

Theorem A.4 (Conic complementarity)**.** *For K a non-empty closed convex cone, there holds*

$$\boldsymbol{y} \in \mathscr{N}_K(\boldsymbol{x}) \iff K \ni \boldsymbol{x} \perp \boldsymbol{y} \in K^\circ \iff \boldsymbol{x} \in \mathscr{N}_{K^\circ}(\boldsymbol{y})$$

where the "$\cdot \perp \cdot$" is used to mean $\langle \cdot, \cdot\rangle = 0$.

Proof. From Theorem A.2 and Properties A.9 and A.15,

$$\begin{aligned}
\boldsymbol{y} \in \mathscr{N}_K(\boldsymbol{x}) &\iff \boldsymbol{y} \in \partial \mathscr{I}_K(\boldsymbol{x}) \\
&\iff \mathscr{I}_K(\boldsymbol{x}) + \mathscr{I}_K^\star(\boldsymbol{y}) = \langle \boldsymbol{x},\boldsymbol{y}\rangle \\
&\iff \mathscr{I}_K(\boldsymbol{x}) + \mathscr{I}_{K^\circ}(\boldsymbol{y}) = \langle \boldsymbol{x},\boldsymbol{y}\rangle
\end{aligned}$$

The equality is only possible on the effective domain of the left-hand side, on which $\mathscr{I}_K(\boldsymbol{x}) + \mathscr{I}_{K^\circ}(\boldsymbol{y})$ is constant and equal to zero. This means

$$\mathscr{I}_K(\boldsymbol{x}) + \mathscr{I}_{K^\circ}(\boldsymbol{y}) = \langle \boldsymbol{x},\boldsymbol{y}\rangle \iff \boldsymbol{x} \in K \text{ and } \boldsymbol{y} \in K^\circ \text{ and } \langle \boldsymbol{x},\boldsymbol{y}\rangle = 0$$

Then, the rightmost equivalence follows from Corollary A.5. □

A.3 Constrained optimization

We consider the minimization problem

$$p = \min_{\boldsymbol{x}\in C} f(\boldsymbol{x}) \tag{A.4}$$

with f closed and convex, and C a closed convex subset of X.

Note that the constrained minimization problem (A.4) is equivalent to the unconstrained one (A.5),

$$p = \min_{x \in X} (f + \mathscr{I}_C)(x) \tag{A.5}$$

We first recall a sufficient condition for the existence of a solution to problem (A.5) (see e.g., Barbu and Precupanu 2012, p. 72, Theorem 2.11, Proposition 2.10 and Remark 2.13).

Theorem A.5 (Existence of a solution)**.** *Let $f : X \to \bar{\mathbb{R}}$ be convex and lower semi-continuous, and $C \subset X$ is convex and closed. If the minimization problem (A.4) is feasible, that is* $\operatorname{dom} f \cap C \neq \emptyset$*, and f is coercive on C, that is,*

$$\lim_{x \in C, \|x\| \to +\infty} f(x) = +\infty,$$

then f attains its global minimum on C.

A.3.1 Optimality conditions

We now derive sufficient conditions for checking that a point $x_0 \in X$ is a solution to (A.4), i.e., $x_0 \in C$ and $f(x_0) = p$.

Theorem A.6 (Fundamental theorem of convex optimization)**.** *Suppose f a proper closed convex function on X and C a closed convex subset of X, such that f is real-valued and continuous on C. The optimality condition of (A.4) reads*

$$\partial f(x_0) \cap -\mathscr{N}_C(x_0) \neq 0.$$

If f is moreover Gâteaux-differentiable on C, the optimality condition becomes

$$(\nabla f)(x_0) \in -\mathscr{N}_C(x_0).$$

Proof. First, if C is empty the problem is not feasible, $\mathscr{N}_C$ is always empty and the equivalence is trivial. Let us now assume C not empty. From Property A.3, the optimality condition of (A.5) and therefore of (A.4) is $0 \in \partial g(x_0)$.

We can apply Theorem A.1 to get $\partial g = \partial \mathscr{I}_C + \partial f$. Expressing $\partial \mathscr{I}_C$ with Property A.9, this means $0 \in \partial f(x_0) + \mathscr{N}_C(x_0)$. If f is Gâteaux-differentiable at x_0, Property A.4 states that its subdifferential contains only its gradient, so the optimality condition becomes $0 \in \{(\nabla f)(x_0)\} + \mathscr{N}_C(x_0)$. □

Corollary A.6. *If X is an Hilbert space, then for $x, y, \alpha \in X \times X \times \mathbb{R}_+^*$,*

$$y \in \mathscr{N}_C(x) \iff \Pi_C(x + \alpha y) = x$$

where Π_C denotes the orthogonal projection on the closed convex subset C of X,

$$\Pi_C(z) = \arg\min_{x \in C} \frac{1}{2} \langle z - x, z - x \rangle$$

Proof. For any $z \in X$, let $f_z(x) := \frac{1}{2} \langle z - x, z - x \rangle$, which a real-valued, closed, convex, and differentiable function on X.

Applying Theorem A.6 to the minimization of f_z on C, we get

$$x_0 = \Pi_C(z) \iff \nabla f_z(x_0) \in -\mathscr{N}_C(x_0) \iff x_0 - z \in -\mathscr{N}_C(x_0).$$

For $z := x_0 + \alpha y$ with $\alpha \in R_*^+$, this reduces to

$$x_0 = \Pi_C(x_0 + \alpha y) \iff \alpha y \in \mathscr{N}_C(x_0) \iff y \in \mathscr{N}_C(x_0).$$

□

Remark A.3. *Combining Theorem A.6 with Corollary A.6, the optimality condition of (A.4) becomes*

$$\Pi_C(\boldsymbol{x}_0 - \alpha(\nabla f)(\boldsymbol{x}_0)) = \boldsymbol{x}_0, \quad \alpha \in \mathbb{R}_+^*,$$

yielding the rationale for the Projected Gradient Descent algorithm.

Corollary A.7 (Karush–Kuhn–Tucker conditions)**.** *The optimality conditions of the minimization problem*

$$\min_{\boldsymbol{x} \in K} f(\boldsymbol{x})$$

with K is a closed convex cone, and f a proper closed convex function on X and real-valued, continuous and Gâteaux-differentiable on K, are

$$K \ni \boldsymbol{x}_0 \perp (\nabla f)(\boldsymbol{x}_0) \in K^\star$$

and are called the first-order Karush–Kuhn–Tucker conditions.

Proof. This is a direct application of Theorem A.6 combined with the equivalences from Property A.9 and Theorem A.4:

$$(\nabla f)(\boldsymbol{x}_0) \in -\mathscr{N}_K(\boldsymbol{x}_0) \iff (\nabla f)(\boldsymbol{x}_0) \in -\partial \mathscr{I}_K(\boldsymbol{x}_0) \iff K \ni \boldsymbol{x}_0 \perp (\nabla f)(\boldsymbol{x}_0) \in K^\star$$

□

A.3.2 Lagrange multipliers

A last application concerns the structure of the normal cone at the boundary of an implicitly-defined convex set.

Property A.16 (Normal cone at the boundary)**.** *Consider a set C defined as $C := \{\boldsymbol{x} \in X,\ F(\boldsymbol{x}) \leq 0\}$ with F a proper closed convex function on X, continuous on C. For $\boldsymbol{x}_0 \in \operatorname{Bd} C$, there always holds*

$$\mathscr{N}_C(\boldsymbol{x}_0) \supset \{\alpha \boldsymbol{g},\ \boldsymbol{g} \in \partial F(\boldsymbol{x}_0),\ \alpha \in \mathbb{R}_+\}, \tag{A.6}$$

and equality is achieved if $\inf F < 0$. It this later case, we also have the equivalence $F(\boldsymbol{x}) = 0 \iff \boldsymbol{x}_0 \in \operatorname{Bd} C$, so that for any $\boldsymbol{x}_0$ such that $F(\boldsymbol{x}) < 0$, $\mathscr{N}_C(\boldsymbol{x}_0) = \{\boldsymbol{0}\}$.

Proof. C is convex (from the convexity of F) and closed (Property A.1). Moreover, as F is continuous on C, $F(\boldsymbol{x}) < 0 \implies \boldsymbol{x} \in \operatorname{int} C$ and therefore $\boldsymbol{x} \in \operatorname{Bd} C \implies F(\boldsymbol{x}) = 0$.

We can easily show that $\{\alpha \boldsymbol{g},\ \boldsymbol{g} \in \partial F(\boldsymbol{x}_0)\ \alpha \in \mathbb{R}_+\} \subset \mathscr{N}_C(\boldsymbol{x}_0)$. Indeed, let $\boldsymbol{x}_0 \in \operatorname{Bd} C$ and $\boldsymbol{g} \in \partial F(\boldsymbol{x}_0)$. This means

$$\begin{aligned} F(\boldsymbol{x}) &\geq F(\boldsymbol{x}_0) + \langle \boldsymbol{x} - \boldsymbol{x}_0, \boldsymbol{y} \rangle && \forall \boldsymbol{x} \in X \\ 0 &\geq \langle \boldsymbol{x} - \boldsymbol{x}_0, \boldsymbol{y} \rangle && \forall \boldsymbol{x},\ F(\boldsymbol{x}) \leq 0 \end{aligned}$$

and therefore $\boldsymbol{y} \in \mathscr{N}_C(\boldsymbol{x}_0)$.

Now assume $\inf F < 0$, and let us prove the converse inclusion. Let $\boldsymbol{y} \in \mathscr{N}_C(\boldsymbol{x}_0)$; if $\boldsymbol{y} = \boldsymbol{0}$, we have $\exists \boldsymbol{g} \in \partial F(\boldsymbol{x}_0), \boldsymbol{0}\boldsymbol{g} = \boldsymbol{y}$ as $\partial F(\boldsymbol{x}_0)$ is not empty (Moreau 1966–1967, paragraph 10.c). Now, let us assume $\boldsymbol{y} \neq \boldsymbol{0}$, and consider the constrained optimization problem

$$\min_{\boldsymbol{x} \in C} -\langle \boldsymbol{x}, \boldsymbol{y} \rangle. \tag{A.7}$$

From Theorem A.6, $\boldsymbol{x}_0$ satisfies the optimality condition of (A.7). Now, suppose that the minimum is also reached at $\boldsymbol{x} \in \operatorname{int} C$; this means $\boldsymbol{y} \in \mathscr{N}_C(\boldsymbol{x}) = \{\boldsymbol{0}\}$ from Property A.10, which is a contradiction. Therefore, $\forall \boldsymbol{x} \in \operatorname{int} C, -\langle \boldsymbol{x}, \boldsymbol{y} \rangle > -\langle \boldsymbol{x}_0, \boldsymbol{y} \rangle$. As $\boldsymbol{x}_0 \in \operatorname{Bd} C$, we get

$$\boldsymbol{x} \in \operatorname{int} C \implies F(\boldsymbol{x}) < \underbrace{F(\boldsymbol{x}_0)}_{0} \implies \langle \boldsymbol{x} - \boldsymbol{x}_0, \boldsymbol{y} \rangle < 0$$

or equivalently

$$\langle \boldsymbol{x} - \boldsymbol{x}_0, \boldsymbol{y} \rangle \geq 0 \implies F(\boldsymbol{x}) \geq F(\boldsymbol{x}_0). \tag{A.8}$$

Let use denote by $\mathscr{H}_{\boldsymbol{y}}(\boldsymbol{x}_0)$ the half-space

$$\mathscr{H}_{\boldsymbol{y}}(\boldsymbol{x}_0) := \{\boldsymbol{x} \in X, \langle \boldsymbol{x} - \boldsymbol{x}_0, \boldsymbol{y} \rangle \geq 0\}.$$

Then Equation (A.8) means that $\boldsymbol{x}_0$ satisfies the optimality conditions of the minimization problem

$$\min_{\boldsymbol{x} \in \mathscr{H}_{\boldsymbol{y}}(\boldsymbol{x}_0)} F(\boldsymbol{x}),$$

which are, using Property (A.3) then Theorem A.1 (F is continuous and both terms are convex and finite at $\boldsymbol{x}_0$),

$$\begin{aligned} \mathbf{0} &\in \partial\Big(F + \mathscr{I}_{\mathscr{H}_{\boldsymbol{y}}(\boldsymbol{x}_0)}\Big)(\boldsymbol{x}_0) = \partial F(\boldsymbol{x}_0) + \mathscr{N}_{\mathscr{H}_{\boldsymbol{y}}(\boldsymbol{x}_0)}(\boldsymbol{x}_0) \\ \mathbf{0} &\in \partial F(\boldsymbol{x}_0) + \{-\alpha \boldsymbol{y}, \alpha \in \mathbb{R}_+\}. \end{aligned}$$

As we supposed that $\inf F < 0$, $\mathbf{0} \notin \partial F(\boldsymbol{x}_0)$, and therefore $\exists \alpha \in R_+$ and $\boldsymbol{g} \in \partial F(\boldsymbol{x}_0)$ such that $\boldsymbol{y} = \alpha \boldsymbol{g}$.

It remains to show that $\boldsymbol{x} \in \operatorname{int} C \implies F(\boldsymbol{x}) < 0$. Suppose $F(\boldsymbol{x}) = 0$. As $\inf F < 0$, $\mathbf{0} \notin \partial F(\boldsymbol{x}_0)$, and we can find a non-zero element $\boldsymbol{y} \in \partial F$, which is non-empty. From the duality of X and Y, there exists $\boldsymbol{x} \in X$, $\langle \boldsymbol{x}, \boldsymbol{y} \rangle > 0$. For every $\epsilon > 0$, let $\boldsymbol{x}_\epsilon = \boldsymbol{x}_0 + \frac{\epsilon}{\|\boldsymbol{x}\|}\boldsymbol{x}$; $\|\boldsymbol{x}_0 - \boldsymbol{x}_\epsilon\| \leq \epsilon$ and $F(\boldsymbol{x}_\epsilon) > 0$, i.e., $\boldsymbol{x}_0 \in \operatorname{Bd} C$. □

The qualification hypothesis $\inf F < 0$ is not necessary, however the equality would then require supplemental constraints on the structure of ∂F. Intuitively, we require F to go to zero with a strictly positive slope in all directions; the problem is indeed similar to that of the existence of Lagrange Multipliers, which require a surjective differential for the application of the Implicit Function Theorem.

An example of a function for which inclusion (A.6) is strict while the subdifferential is nowhere equal to $\{\mathbf{0}\}$ is

$$F : \mathbb{R} \to \mathbb{R},\ x \mapsto \begin{cases} x^2 & \text{if } x \leq 0 \\ x & \text{if } x > 0. \end{cases}$$

Indeed, $\partial F(0) = [0, 1]$ while $\partial \mathscr{I}_C = \mathbb{R}$.

Conversely, an example for which $\min F = 0$ yet the equality holds is the case of affine equality constraints:

Property A.17. *Suppose $A : X \to E$, $\boldsymbol{x} \mapsto L\boldsymbol{x} + \boldsymbol{e}$ an affine map between two Hilbert spaces. Let $F : \boldsymbol{x} \mapsto \|A(\boldsymbol{x})\|$, and $C := \{\boldsymbol{x} \in X, F(\boldsymbol{x}) = 0\}$. Then for $\boldsymbol{x}_0$ such that $A(\boldsymbol{x}_0) = \mathbf{0}$,*

$$\mathscr{N}_C(\boldsymbol{x}_0) = \operatorname{Im} L^\intercal = \{\alpha \boldsymbol{g},\ \boldsymbol{g} \in \partial F(\boldsymbol{x}_0),\ \alpha \in \mathbb{R}_+\}.$$

Proof. First, suppose $\boldsymbol{y} \in \mathscr{N}_C(\boldsymbol{x}_0)$. $\forall \boldsymbol{x} \in C, \langle \boldsymbol{x} - \boldsymbol{x}_0, \boldsymbol{y} \rangle \leq 0$. As $A(\boldsymbol{x}_0) = 0$, $F(\boldsymbol{x}) = 0 \iff A(\boldsymbol{x}) - A(\boldsymbol{x}_0) = \mathbf{0} \iff L(\boldsymbol{x} - \boldsymbol{x}_0) = \mathbf{0}$. Therefore, $\boldsymbol{y}$ must obey that $\forall \boldsymbol{x} \in \operatorname{Ker} L$, $\langle \boldsymbol{x}, \boldsymbol{y} \rangle \leq 0$, which means $\boldsymbol{y} \in (\operatorname{Ker} L)^\perp$. We thus have $\mathscr{N}_C(\boldsymbol{x}_0) \subset \operatorname{Im} L^\intercal$.

Now, let us prove that

$$\operatorname{Im} L^\intercal \subset \{\alpha \boldsymbol{g},\ \boldsymbol{g} \in \partial F(\boldsymbol{x}_0),\ \alpha \in \mathbb{R}_+\},$$

which will achieve the demonstration, the converse inclusion being granted by Property A.16. Property A.12 states that we always have $\partial F(\boldsymbol{x}_0) \supset L^\intercal \partial \|\cdot\|(A(\boldsymbol{x}_0))$. Since $A(\boldsymbol{x}_0) = 0$, using Theorem (A.2),

$$\boldsymbol{g} \in \partial \|\cdot\|(A(\boldsymbol{x}_0)) \iff \mathbf{0} \in \partial \|\cdot\|^*(\boldsymbol{g}).$$

Direct computations yield $\|\cdot\|^* = \mathscr{I}_{\mathscr{B}(\mathbf{0},1)}$, the characteristic function of the unit closed ball centered at $\mathbf{0}$. This means

$$\boldsymbol{g} \in \partial \|\cdot\|(A(\boldsymbol{x}_0)) \iff \mathbf{0} \in \mathscr{N}_{\mathscr{B}(\mathbf{0},1)}(\boldsymbol{g}) \iff \|\boldsymbol{g}\| \leq 1.$$

It results that $\partial F(\boldsymbol{x}_0) \supset L^\intercal \mathscr{B}(\mathbf{0},1)$. Then, for all $\boldsymbol{y} \in \operatorname{Im} L^\intercal$, $\exists \alpha \in \mathbb{R}_+$, $\boldsymbol{y} = L^\intercal \alpha \boldsymbol{g}$ with $\|\boldsymbol{g}\| \leq 1$, and in particular, $(L^\intercal \boldsymbol{g}) \in \partial F(\boldsymbol{x}_0)$.

□

Property A.18 (KKT multiplier)**.** *Let $C := \{\boldsymbol{x} \in X, F(\boldsymbol{x}) \leq 0$ and $A(\boldsymbol{x}) = 0\}$ with F convex, closed and continuous on C, and A an affine map, $A : X \to E$, $\boldsymbol{x} \mapsto L\boldsymbol{x} + \boldsymbol{e}$. We consider the optimization problem (A.4),*

$$\min_{\boldsymbol{x} \in C} f(\boldsymbol{x}), \tag{A.4}$$

for f closed, convex, continuous and real-valued on C.

If there exists $\boldsymbol{x} \in X$ such that $A(\boldsymbol{x}) = 0$ and $F(\boldsymbol{x}) < 0$, then the optimality conditions of (A.4) read

$$\begin{cases} \emptyset \neq \partial f(\boldsymbol{x}_0) \cap (-\lambda \partial F(\boldsymbol{x}_0) + L^\intercal \boldsymbol{r}) \\ \mathbf{0} = A(\boldsymbol{x}_0) \\ 0 \leq F(\boldsymbol{x}_0) \perp \lambda \geq 0. \end{cases} \tag{A.9}$$

$\lambda \in \mathbb{R}$ is the Karush–Kuhn–Tucker multiplier associated to the constraint $F(\boldsymbol{x}) \leq 0$, and $\boldsymbol{r} \in E^\star$ is the vector of Lagrange multipliers associated to the constraint $A(\boldsymbol{x}) = 0$.

Proof. Property A.6 gives the optimality condition $\partial f(\boldsymbol{x}_0) \cap \mathscr{N}_C \neq 0$. Let us express $\mathscr{N}_C = \mathscr{N}_{C_A \cap C_F}$, $C_A := \{A(\boldsymbol{x}) = \mathbf{0}\}$, $C_F := \{F(\boldsymbol{x}) \leq 0\}$.

Our hypothesis, $\exists \boldsymbol{x}, A(\boldsymbol{x}) = \mathbf{0}$ and $F(\boldsymbol{x}) < 0$, implies that $C_A \cap \operatorname{int} C_F \neq \emptyset$; we can therefore use the Corollary A.1 on the subdifferential of a sum, and obtain with Property A.9 that $\mathscr{N}_{C_A \cap C_F} = \mathscr{N}_{C_A} + \mathscr{N}_{C_F}$.

$\mathscr{N}_{C_A}$ is straightforward to compute, and Property A.16 gives us the expression of the normal cone $\mathscr{N}_{C_F}(\boldsymbol{x}_0)$,

$$\mathscr{N}_{C_A}(\boldsymbol{x}_0) = \begin{cases} \emptyset & \text{if } A(\boldsymbol{x}_0) \neq \mathbf{0} \\ \operatorname{Im} L^\intercal & \text{if } A(\boldsymbol{x}_0) = \mathbf{0} \end{cases}$$

$$\mathscr{N}_{C_F}(\boldsymbol{x}_0) = \begin{cases} \emptyset & \text{if } F(\boldsymbol{x}_0) > 0 \\ \{\mathbf{0}\} & \text{if } F(\boldsymbol{x}_0) < 0 \\ \{\lambda \boldsymbol{g},\ \lambda \in \mathbb{R}_+,\ \boldsymbol{g} \in \partial F(\boldsymbol{x}_0)\} & \text{if } F(\boldsymbol{x}_0) = 0. \end{cases}$$

□

Remark A.4. *Property A.18 can easily be extended to the case of multiple constraints thanks to Property A.6, with $F := \max_{i=1\dots n} F_i(\boldsymbol{x})$. The optimality conditions are then, for f Gâteaux-differentiable, and in the absence of linear constraints,*

$$\begin{cases} (\nabla f)(\boldsymbol{x}_0) = \displaystyle\sum_{i=1}^{n} \lambda_i \boldsymbol{g}_i,\ \boldsymbol{g}_i \in \partial F_i \\ \mathbb{R}^n_- \ni (F_i(\boldsymbol{x}_0)) \perp (\lambda_i) \in \mathbb{R}^n_+. \end{cases}$$

Theorem A.7 (Lagrange duality)**.** *With the notations of Property A.18, let p be the minimum of the constrained minimization problem (A.4),*

$$p = \min_{\boldsymbol{x} \in C} f(\boldsymbol{x}). \tag{A.4}$$

Under the qualification hypothesis of A.18, if there exists $\boldsymbol{x}_0 \in C$ such that $p = f(\boldsymbol{x}_0)$, then there exists $(\lambda_0, \boldsymbol{r}_0) \in \mathbb{R}_+ \times E$ such that the tuple $(\boldsymbol{x}_0, \lambda_0, \boldsymbol{r}_0)$ is a saddle points of the Lagrangian $\mathscr{L}(\boldsymbol{x}, \lambda, \boldsymbol{r})$,

$$\mathscr{L}(\boldsymbol{x}, \lambda, \boldsymbol{r}) := f(\boldsymbol{x}) + \lambda F(\boldsymbol{x}) + \langle \boldsymbol{r}, A(\boldsymbol{x}) \rangle .$$

That is, $(\lambda_0, \boldsymbol{r}_0)$ is a solution to the dual maximization problem,

$$d = \max_{\boldsymbol{r} \in E, \lambda \geq 0} J(\lambda, \boldsymbol{r}), \quad J(\lambda, \boldsymbol{r}) := \min_{\boldsymbol{x} \in X} (\mathscr{L}(\boldsymbol{x}, \lambda, \boldsymbol{r})). \tag{A.10}$$

with $J(\lambda_0, \boldsymbol{r}_0) = \mathscr{L}(\boldsymbol{x}_0, \lambda_0, \boldsymbol{r}_0)$.

Moreover, $p = d$*, and if* $(\lambda_0, \boldsymbol{r}_0) \in \mathbb{R}_+ \times E^\star$ *is a solution to the dual problem (A.10), any solution* $\boldsymbol{x}_0 \in X$ *to the primal problem (A.4) must satisfy* $d = J(\lambda_0, \boldsymbol{r}_0) = \mathscr{L}(\boldsymbol{x}_0, \lambda_0, \boldsymbol{r}_0)$.

Proof. First, let us show that J is concave (i.e., $-J$ is convex). For $\alpha \in]0,1[$,

$$\begin{aligned}
\alpha J(\lambda_1, \boldsymbol{r}_1) + (1-\alpha) J(\lambda_2, \boldsymbol{r}_2) &= \min_{\boldsymbol{x} \in X} (\alpha \mathscr{L}(\boldsymbol{x}, \lambda_1, \boldsymbol{r}_1)) + \min_{\boldsymbol{x} \in X} ((1-\alpha) \mathscr{L}(\boldsymbol{x}, \lambda_2, \boldsymbol{r}_2)) \\
&\geq \min_{\boldsymbol{x} \in X} (\alpha \mathscr{L}(\boldsymbol{x}, \lambda_1, \boldsymbol{r}_1) + (1-\alpha) \mathscr{L}(\boldsymbol{x}, \lambda_2, \boldsymbol{r}_2)) \\
&\geq \min_{\boldsymbol{x} \in X} \mathscr{L}(\boldsymbol{x}, \alpha\lambda_1 + (1-\alpha)\lambda_2, \alpha \boldsymbol{r}_1 + (1-\alpha)\boldsymbol{r}_2) \\
&\geq J(\alpha\lambda_1 + (1-\alpha)\lambda_2, \alpha \boldsymbol{r}_1 + (1-\alpha)\boldsymbol{r}_2)
\end{aligned}$$

This means that we can use Property A.6 to get an optimality condition for the dual problem (A.10),

$$\emptyset \neq -\partial(-J)(\lambda, \boldsymbol{r}) \neq \left(\mathscr{N}_{\mathbb{R}_+}(\lambda) \times \boldsymbol{0}_{E^\star}\right). \tag{A.11}$$

Suppose that $(\boldsymbol{x}_0)$ is a solution to the primal problem (A.4). From Property (A.18), this means that there exists $(\lambda_0, \boldsymbol{r}_0)$ such that

$$\begin{cases} \emptyset \neq \partial f(\boldsymbol{x}_0) \cap (-\lambda_0 \partial F(\boldsymbol{x}_0) + L^\intercal \boldsymbol{r}_0) \\ \boldsymbol{0} = A(\boldsymbol{x}_0) \\ 0 \leq F(\boldsymbol{x}_0) \perp \lambda_0 \geq 0. \end{cases} \tag{A.9}$$

The first equation ensures that $\boldsymbol{x}_0$ is a solution to the minimization problem $J(\lambda_0, \boldsymbol{r}_0)$, i.e., $J(\lambda_0, \boldsymbol{r}_0) = \mathscr{L}(\boldsymbol{x}_0, \lambda_0, \boldsymbol{r}_0)$. It remains to show that $(\lambda_0, \boldsymbol{r}_0)$ satisfies the optimality condition (A.11) of the dual problem.

As J is a minimum over $\boldsymbol{x} \in X$, $\forall \lambda, \boldsymbol{r} \in \mathbb{R} \times E$,

$$\begin{aligned}
J(\lambda, \boldsymbol{r}) &\leq \mathscr{L}(\boldsymbol{x}_0, \lambda, \boldsymbol{r}) \\
&\leq F(\boldsymbol{x}_0) + \lambda F(\boldsymbol{x}_0) + \langle \boldsymbol{r}, A(\boldsymbol{x}_0) \rangle \\
&\leq J(\boldsymbol{x}_0) + (\lambda - \lambda_0) F(\boldsymbol{x}_0) + \langle \boldsymbol{r} - \boldsymbol{r}_0, A(\boldsymbol{x}_0) \rangle
\end{aligned}$$

which by definition means that $(F(\boldsymbol{x}_0), A(\boldsymbol{x}_0)) \in -\partial(-J)$. We can easily check from the conditions (A.9) that $(F(\boldsymbol{x}_0), A(\boldsymbol{x}_0)) \in \mathscr{N}_{\mathbb{R}_+}(\lambda_0) \times \boldsymbol{0}_{E^\star}$, meaning that the optimality conditions of the dual problem (A.10) are satisfied.

Notice that $\lambda_0 F(\boldsymbol{x}_0) = 0$ and $\langle \boldsymbol{r}_0, A(\boldsymbol{x}_0) \rangle = 0$, i.e., $\mathscr{L}(\boldsymbol{x}_0, \lambda_0, \boldsymbol{r}_0) = f(\boldsymbol{x}_0)$, and therefore $p = d$.

We have proved that if there exists a solution to the primal problem (A.4), then $p = d$. This is actually always true, as we will show below; but for now, assume that $p = d$, and let us show that, for $(\lambda_0, \boldsymbol{r}_0)$ solution to the dual problem (A.10), any primal solution $\boldsymbol{x}_0$ is such that $\mathscr{L}(\lambda_0, \boldsymbol{r}_0, \boldsymbol{x}_0) = J(\lambda_0, \boldsymbol{r}_0)$. Indeed, as $\boldsymbol{x}_0 \in C_F$, $\mathscr{L}(\boldsymbol{x}_0, \lambda_0, \boldsymbol{r}_0) = f(\boldsymbol{x}_0) + \lambda_0 F(\boldsymbol{x}_0)$, with $\lambda_0 F(\boldsymbol{x}_0) \leq 0$. Therefore

$$d = J(\lambda_0, \boldsymbol{r}_0) \leq \mathscr{L}(\boldsymbol{x}_0, \lambda_0, \boldsymbol{r}_0) \leq f(\boldsymbol{x}_0) = p = d,$$

from which we deduce that $\mathscr{L}(\boldsymbol{x}_0, \lambda_0, \boldsymbol{r}_0) = J(\lambda_0, \boldsymbol{r}_0)$.

In the remainder of this proof, we show that $p = d$ regardless of the existence of a primal solution. It always hold that

$$d = \max_{\lambda \geq 0, \boldsymbol{r} \in E} \min_{\boldsymbol{x} \in X} \mathscr{L}(\boldsymbol{x}, \lambda, \boldsymbol{r}) \leq \max_{\lambda \geq 0, \boldsymbol{r} \in E} \min_{\boldsymbol{x} \in C} \mathscr{L}(\boldsymbol{x}, \lambda, \boldsymbol{r}) \leq \max_{\lambda \geq 0, \boldsymbol{r} \in E} \min_{\boldsymbol{x} \in C} f(\boldsymbol{x}) \leq p.$$

We therefore have to show that $p \leq d$. Let us introduce the affine map $\hat{A} : X \to X \times E$, $\boldsymbol{x} \mapsto \begin{bmatrix} \boldsymbol{x} \\ A(\boldsymbol{x}) + \boldsymbol{e} \end{bmatrix}$, and the characteristic function $g : X \times E \to R$, $(\boldsymbol{x}, \boldsymbol{y}) \mapsto \mathscr{I}_{C_F \times \{\boldsymbol{0}_E\}}$.

We can use Fenchel's duality theorem (A.3) to get that

$$\begin{aligned}
p &= \inf_{\boldsymbol{x}\in X}\left(f(\boldsymbol{x})+g\circ\hat{A}\boldsymbol{g}\right) = \sup_{\boldsymbol{y}\in Y,\boldsymbol{r}\in E^\star} J_f(\boldsymbol{r},\boldsymbol{y}),\\
J_f(\boldsymbol{r},\boldsymbol{y}) &= -f^\star(\boldsymbol{y}+L^\intercal\boldsymbol{r})-g^\star(-\boldsymbol{y},-\boldsymbol{r})-\langle \boldsymbol{e},\boldsymbol{r}\rangle\\
&= \sup_{\boldsymbol{y}\in Y,\boldsymbol{r}\in E^\star} -\sup_{\boldsymbol{x}\in X}\left(\langle \boldsymbol{x},\boldsymbol{y}+L^\intercal\boldsymbol{r}\rangle - f(\boldsymbol{x})\right)-g^\star(-\boldsymbol{y},-\boldsymbol{r})-\langle \boldsymbol{e},\boldsymbol{r}\rangle\\
&= \sup_{\boldsymbol{y}\in Y,\boldsymbol{r}\in E^\star} -\sup_{\boldsymbol{x}\in X}\left(\langle \boldsymbol{x},\boldsymbol{y}\rangle + \langle A(\boldsymbol{x}),\boldsymbol{r}\rangle - f(\boldsymbol{x})\right)-g^\star(-\boldsymbol{y},-\boldsymbol{r})\\
&= \sup_{\boldsymbol{y}\in Y,\boldsymbol{r}\in E^\star} \inf_{\boldsymbol{x}\in X}\left(\langle \boldsymbol{x},\boldsymbol{y}\rangle + \langle A(\boldsymbol{x}),\boldsymbol{r}\rangle + f(\boldsymbol{x})\right)-g^\star(\boldsymbol{y},\boldsymbol{r}).
\end{aligned}$$

As p is a supremum, we need only consider the set of $\boldsymbol{y},\boldsymbol{r}$ for which $g^\star(\boldsymbol{y},\boldsymbol{r})<+\infty$. Let us express $g^\star$:

$$\begin{aligned}
g^\star(\boldsymbol{y},\boldsymbol{r}) &= \sup_{\boldsymbol{x}\in X,\boldsymbol{z}\in E}\left(\langle \boldsymbol{y},\boldsymbol{x}\rangle - \mathscr{I}_{C_F}(\boldsymbol{x})+\langle \boldsymbol{r},\boldsymbol{z}\rangle - \mathscr{I}_{\boldsymbol{0}_E}(\boldsymbol{z})\right)\\
&= \sup_{\boldsymbol{x}\in X}\left(\langle \boldsymbol{y},\boldsymbol{x}\rangle - \mathscr{I}_{C_F}(\boldsymbol{x})\right)+\sup_{\boldsymbol{r}\in E}\left(\langle \boldsymbol{r},\boldsymbol{z}\rangle - \mathscr{I}_{\boldsymbol{0}_E}(\boldsymbol{z})\right)\\
&= -\inf_{\boldsymbol{x}\in C_F}\left(-\langle \boldsymbol{y},\boldsymbol{x}\rangle\right).
\end{aligned}$$

The optimality condition for $\inf_{\boldsymbol{x}\in C_F}(-\langle \boldsymbol{y},\boldsymbol{x}\rangle)$ is $\exists \boldsymbol{x}_0\in X,\ \boldsymbol{y}\in\mathscr{N}_{C_F}(\boldsymbol{x}_0)$. Moreover, if $g^\star(\boldsymbol{y},\boldsymbol{r})$ is finite but the optimum is non attained, then $\forall\epsilon>0$, the penalized optimization problem

$$q_\epsilon(\boldsymbol{y}) = \min_{\boldsymbol{x}\in C_F}\left(-\langle \boldsymbol{y},\boldsymbol{x}\rangle+\epsilon\|\boldsymbol{x}\|\right) \tag{A.12}$$

attains its minimum, and $\lim_{\epsilon\to 0_+} q_\epsilon(\boldsymbol{y}) = -g^\star(\boldsymbol{y},\boldsymbol{r})$. Indeed, the function $h_\epsilon : X\to\bar{\mathbb{R}}$,

$$h_\epsilon : \boldsymbol{x}\mapsto -\langle \boldsymbol{y},\boldsymbol{x}\rangle+\epsilon\|\boldsymbol{x}\|$$

is convex, lower semi-continuous and coercive on a reflexive Banach space, and therefore, from Theorem A.5, attains its minimum on C_F. Moreover $\forall\delta>0$, $\exists \boldsymbol{x}_0\in C_F$, $\langle \boldsymbol{x}_0,\boldsymbol{y}\rangle > g^\star(\boldsymbol{y},\boldsymbol{r})-\delta$, therefore $\exists\epsilon_0$ such that $\forall\epsilon<\epsilon_0$, $\langle \boldsymbol{x}_0,\boldsymbol{y}\rangle-\epsilon\|\boldsymbol{x}_0\| > g^\star(\boldsymbol{y},\boldsymbol{r})-\delta$. i.e., $\lim_{\epsilon\to 0_+}(q_\epsilon(\boldsymbol{y})+g^\star(\boldsymbol{y},\boldsymbol{r}))<\delta$. Since there also holds $q_\epsilon\geq -g^\star(\boldsymbol{y},\boldsymbol{r})$, $\lim_{\epsilon\to 0_+}(q_\epsilon(\boldsymbol{y})+g^\star(\boldsymbol{y}))=0$.

Let us denote by V the subset of Y for which $g^\star(\boldsymbol{y},\boldsymbol{r})$ is finite. For any $\boldsymbol{y}\in V$, for any $\epsilon>0$, we note $\boldsymbol{x}_\epsilon$ the element of C_F at which the optimal value of q_ϵ is reached. We have

$$-g^\star(\boldsymbol{y},\boldsymbol{r}) \leq \lim_{\epsilon\to 0_+}\langle \boldsymbol{x}_\epsilon,-\boldsymbol{y}\rangle \leq \lim_{\epsilon\to 0_+}\left(\langle \boldsymbol{x}_\epsilon,-\boldsymbol{y}\rangle+\epsilon\|\boldsymbol{x}_\epsilon\|\right) = \lim_{\epsilon\to 0_+} q_\epsilon(\boldsymbol{y}) = -g^\star(\boldsymbol{y},\boldsymbol{r}),$$

therefore $\lim_{\epsilon\to 0_+}\epsilon\|\boldsymbol{x}_\epsilon\|=0$. As for all $\boldsymbol{y}_\epsilon\in\mathscr{B}(\boldsymbol{y},\epsilon)$,

$$\lim_{\epsilon\to 0_+}\left(\langle \boldsymbol{x}_\epsilon,-\boldsymbol{y}\rangle+\epsilon\|\boldsymbol{x}_\epsilon\|\right) \geq \lim_{\epsilon\to 0_+}\left\langle \boldsymbol{x}_\epsilon,-\boldsymbol{y}_\epsilon\right\rangle \geq \lim_{\epsilon\to 0_+}\left(\langle \boldsymbol{x}_\epsilon,-\boldsymbol{y}\rangle-\epsilon\|\boldsymbol{x}_\epsilon\|\right),$$

we deduce that

$$\lim_{\epsilon\to 0_+}\left\langle \boldsymbol{x}_\epsilon,-\boldsymbol{y}_\epsilon\right\rangle = \lim_{\epsilon\to 0_+}\left(\langle \boldsymbol{x}_\epsilon,-\boldsymbol{y}\rangle\right) = -g^\star(\boldsymbol{y},\boldsymbol{r}).$$

Now, the optimality condition of (A.12) is $\exists \boldsymbol{x}_\epsilon\in X,\ \boldsymbol{y}\in\mathscr{B}(\boldsymbol{0},\epsilon)+\mathscr{N}_{C_F}(\boldsymbol{x}_\epsilon)$. Going back to our original problem, we can now write that $\forall\boldsymbol{y}\in V$, $\forall\boldsymbol{r}\in E^\star$,

$$\begin{aligned}
J_f(\boldsymbol{y},\boldsymbol{r}) &= \lim_{\epsilon\to 0_+}\inf_{\boldsymbol{x}\in X}\left(\langle \boldsymbol{x},\boldsymbol{y}\rangle+\langle A(\boldsymbol{x}),\boldsymbol{r}\rangle+f(\boldsymbol{x})\right)-\left\langle \boldsymbol{x}_\epsilon,\boldsymbol{y}_\epsilon\right\rangle && \boldsymbol{y}_\epsilon\in\mathscr{N}_{C_F}(\boldsymbol{x}_\epsilon)\\
&= \lim_{\epsilon\to 0_+}\inf_{\boldsymbol{x}\in X}\left(\left\langle \boldsymbol{x}-\boldsymbol{x}_\epsilon,\boldsymbol{y}_\epsilon\right\rangle+\left\langle \boldsymbol{x},\boldsymbol{y}-\boldsymbol{y}_\epsilon\right\rangle+\langle A(\boldsymbol{x}),\boldsymbol{r}\rangle+f(\boldsymbol{x})\right) && \boldsymbol{y}_\epsilon\in\mathscr{N}_{C_F}(\boldsymbol{x}_\epsilon)\\
&\leq \lim_{\epsilon\to 0_+}\inf_{\boldsymbol{x}\in X}\left(\left\langle \boldsymbol{x}-\boldsymbol{x}_\epsilon,\boldsymbol{y}_\epsilon\right\rangle+\epsilon\|\boldsymbol{x}\|+\langle A(\boldsymbol{x}),\boldsymbol{r}\rangle+f(\boldsymbol{x})\right) && \boldsymbol{y}_\epsilon\in\mathscr{N}_{C_F}(\boldsymbol{x}_\epsilon)
\end{aligned}$$

From Property A.16, we know that

$$\mathscr{N}_{C_F}(\boldsymbol{x}) = \{\boldsymbol{y}\in Y, \boldsymbol{y}=\alpha\boldsymbol{g},\boldsymbol{g}\in\partial F(\boldsymbol{x}) \text{ and } 0\leq\alpha\perp F(\boldsymbol{x})\geq 0\}.$$

Therefore, for $\boldsymbol{y}_\epsilon \in \mathcal{N}_{C_F}(\boldsymbol{x}_\epsilon)$, and $\forall \boldsymbol{x} \in X$,

$$\left\langle \boldsymbol{x} - \boldsymbol{x}_\epsilon, \boldsymbol{y}_\epsilon \right\rangle = \langle \boldsymbol{x} - \boldsymbol{x}_\epsilon, \alpha \boldsymbol{g} \rangle + \alpha F(\boldsymbol{x}_\epsilon) \leq \alpha F(\boldsymbol{x})$$

as $\boldsymbol{g} \in \partial F(\boldsymbol{x}_\epsilon)$, and then

$$\begin{aligned} J_f(\boldsymbol{y}, \boldsymbol{r}) &\leq \lim_{\epsilon \to 0_+} \sup_{\alpha \in R_+} \inf_{\boldsymbol{x} \in X} (\alpha F(\boldsymbol{x}) + \epsilon \|\boldsymbol{x}\| + \langle A(\boldsymbol{x}), \boldsymbol{r} \rangle + f(\boldsymbol{x})) \\ &\leq \sup_{\alpha \in R_+} \inf_{\boldsymbol{x} \in X} (\alpha F(\boldsymbol{x}) + \langle A(\boldsymbol{x}), \boldsymbol{r} \rangle + f(\boldsymbol{x})) \\ &\leq \sup_{\alpha \in R_+} J(\alpha, \boldsymbol{r}). \end{aligned}$$

We conclude that

$$p = \sup_{\boldsymbol{y} \in Y, \boldsymbol{r} \in E^\star} J_f(\boldsymbol{y}, \boldsymbol{r}) = \sup_{\boldsymbol{y} \in V, \boldsymbol{r} \in E^\star} J_f(\boldsymbol{y}, \boldsymbol{r}) \leq \sup_{\boldsymbol{r} \in E^\star} \sup_{\alpha \in R_+} J(\alpha, \boldsymbol{r}) \leq d.$$

□

B Discrete Coulomb Friction Problem solvers

B.1 Newton SOC Fischer–Burmeister function

B.1.1 SOC Fischer–Burmeister derivatives

Let us recall that for any pair of vectors $(\hat{\boldsymbol{r}}, \hat{\boldsymbol{u}}) \in \mathbb{R}^d \times \mathbb{R}^d$, the vector $\boldsymbol{z}^2 := \hat{\boldsymbol{r}} \circ \hat{\boldsymbol{r}} + \hat{\boldsymbol{u}} \circ \hat{\boldsymbol{u}}$ is in the SOC $\mathscr{K}_1$, and its square root can be computed as $\boldsymbol{z} = \sqrt{\lambda_1}\boldsymbol{v}_1 + \sqrt{\lambda_2}\boldsymbol{v}_2$,

$$\begin{aligned}
\boldsymbol{w} &:= \hat{\boldsymbol{r}}_{\mathrm{N}}\hat{\boldsymbol{r}}_{\mathbf{T}} + \hat{\boldsymbol{u}}_{\mathrm{N}}\hat{\boldsymbol{u}}_{\mathbf{T}} \\
\lambda_i &= \|\hat{\boldsymbol{r}}\|^2 + \|\hat{\boldsymbol{u}}\|^2 + (-1)^i\|2\boldsymbol{w}\| \\
\boldsymbol{v}_i &= \frac{1}{2}\begin{cases} \left(1; (-1)^i \dfrac{\boldsymbol{w}}{\|\boldsymbol{w}\|}\right) & \text{if } \boldsymbol{w} \neq \mathbf{0} \\ \left(1; (-1)^i \boldsymbol{e}\right) & \text{if } \boldsymbol{w} = \mathbf{0}. \end{cases}
\end{aligned}$$

The SOC Fischer–Burmeister function can thus be evaluated as $f_{\mathrm{FB}}^{\mathscr{K}}(\hat{\boldsymbol{r}}, \hat{\boldsymbol{u}}) = \sqrt{\lambda_1}\boldsymbol{v}_1 + \sqrt{\lambda_2}\boldsymbol{v}_2 - \hat{\boldsymbol{r}} - \hat{\boldsymbol{u}}$. To compute the gradient of $f_{\mathrm{FB}}^{\mathscr{K}}$, let us recall two properties of the SOC algebra (Chen and Tseng 2005, Property 1):

- If $\boldsymbol{x} \in \operatorname{int}\mathscr{K}_1$, $\boldsymbol{x}$ is invertible[1] and $\boldsymbol{x}^{-1} \in \operatorname{int}\mathscr{K}_1$;
- $\boldsymbol{x} \in \mathbb{R} \times \mathbb{R}^{d-1}$ is invertible $\iff \det \boldsymbol{x} = \boldsymbol{x}_{\mathrm{N}}^2 - \|\boldsymbol{x}_{\mathbf{T}}\|^2 \neq 0$.

Since we always have $\boldsymbol{z} \in \mathscr{K}_1$ and $\det \boldsymbol{z} = \sqrt{\lambda_1\lambda_2}$, we get $\boldsymbol{z}$ invertible $\iff \boldsymbol{z} \in \operatorname{int}\mathscr{K}_1 \iff \lambda_1\lambda_2 \neq 0$. We can then distinguish three cases We have (Chen and Tseng 2005, Proposition 1):

- If $\lambda_2 = 0$, then $\hat{\boldsymbol{r}} = \hat{\boldsymbol{u}} = \mathbf{0}$, and thus $f_{\mathrm{FB}}^{\mathscr{K}}(\hat{\boldsymbol{r}}, \hat{\boldsymbol{u}}) = \mathbf{0}$; we have already found a solution to our problem.
- Otherwise, if $\lambda_1 = 0$, then $\boldsymbol{z} \in \operatorname{Bd}\mathscr{K}_1 \setminus \mathbf{0}$, and we can arbitrarily choose one element of the generalized Jacobian of $f_{\mathrm{FB}}^{\mathscr{K}}$. We take[2]:

$$\frac{\partial f_{\mathrm{FB}}^{\mathscr{K}}}{\partial \hat{\boldsymbol{r}}}(\hat{\boldsymbol{r}}, \hat{\boldsymbol{u}}) = \left(\frac{\hat{\boldsymbol{r}}_{\mathrm{N}}}{\sqrt{\hat{\boldsymbol{r}}_{\mathrm{N}}^2 + \hat{\boldsymbol{u}}_{\mathrm{N}}^2}} - 1\right)\mathbb{I} \text{ and } \frac{\partial f_{\mathrm{FB}}^{\mathscr{K}}}{\partial \hat{\boldsymbol{u}}}(\hat{\boldsymbol{r}}, \hat{\boldsymbol{u}}) = \left(\frac{\hat{\boldsymbol{u}}_{\mathrm{N}}}{\sqrt{\hat{\boldsymbol{r}}_{\mathrm{N}}^2 + \hat{\boldsymbol{u}}_{\mathrm{N}}^2}} - 1\right)\mathbb{I}.$$

- Otherwise, we have $\boldsymbol{z} \in \operatorname{int}\mathscr{K}_1$, and both $\boldsymbol{z}^{-1}$ and the Jacobian of $f_{\mathrm{FB}}^{\mathscr{K}}$ are uniquely defined. For $\boldsymbol{x} \in \mathbb{R}^d$, let us denote $L_{\boldsymbol{x}}$ the $d \times d$ matrix that satisfies $\forall \boldsymbol{y} \in \mathbb{R}^d\ L_{\boldsymbol{x}}\boldsymbol{y} = \boldsymbol{x} \circ \boldsymbol{y}$.

$$L_{\boldsymbol{z}^{-1}} = \frac{1}{\sqrt{\lambda_1\lambda_2}}\begin{pmatrix} \boldsymbol{z}_{\mathrm{N}} & -\boldsymbol{z}_{\mathbf{T}}^\top \\ -\boldsymbol{z}_{\mathbf{T}} & \frac{\sqrt{\lambda_1\lambda_2}}{\boldsymbol{z}_{\mathrm{N}}}\mathbb{I}_{d-1} + \frac{\boldsymbol{z}_{\mathbf{T}}\boldsymbol{z}_{\mathbf{T}}^\top}{\boldsymbol{z}_{\mathrm{N}}} \end{pmatrix}.$$

$$\text{Then} \quad \frac{\partial f_{\mathrm{FB}}^{\mathscr{K}}}{\partial \hat{\boldsymbol{r}}}(\hat{\boldsymbol{r}}, \hat{\boldsymbol{u}}) = \mathbb{I} - L_{\boldsymbol{z}^{-1}}L_{\hat{\boldsymbol{r}}} \text{ and } \frac{\partial f_{\mathrm{FB}}^{\mathscr{K}}}{\partial \hat{\boldsymbol{u}}}(\hat{\boldsymbol{r}}, \hat{\boldsymbol{u}}) = \mathbb{I} - L_{\boldsymbol{z}^{-1}}L_{\hat{\boldsymbol{u}}}.$$

Finally, using the chain rule,

$$\frac{df_{\mathrm{FB}}^{\mathscr{K}}}{d\boldsymbol{r}}(\boldsymbol{r}) = \frac{\partial f_{\mathrm{FB}}^{\mathscr{K}}}{\partial \hat{\boldsymbol{r}}}(\hat{\boldsymbol{r}}, \hat{\boldsymbol{u}})\frac{d\hat{\boldsymbol{r}}}{d\boldsymbol{r}} + \frac{\partial f_{\mathrm{FB}}^{\mathscr{K}}}{\partial \hat{\boldsymbol{u}}}(\hat{\boldsymbol{r}}, \hat{\boldsymbol{u}})\frac{d\hat{\boldsymbol{u}}}{d\tilde{\boldsymbol{u}}}\frac{d\tilde{\boldsymbol{u}}}{d\boldsymbol{u}}\frac{d\boldsymbol{u}}{d\boldsymbol{r}}.$$

Note that $\frac{d\tilde{\boldsymbol{u}}}{d\boldsymbol{u}}$ is not strictly defined when $\boldsymbol{u}_{\mathbf{T}} = \mathbf{0}$, because $\|\boldsymbol{u}_{\mathbf{T}}\|$ is not differentiable at $\mathbf{0}$. We can however take any element of its generalized Jacobian, such as I_d.

[1] With respect to the Jordan product, i.e., $\exists \boldsymbol{y}$ s.t. $\boldsymbol{x} \circ \boldsymbol{y} = \boldsymbol{y} \circ \boldsymbol{x} = [1, \mathbf{0}]$.

[2] $\hat{\boldsymbol{r}}_{\mathrm{N}}^2 + \hat{\boldsymbol{u}}_{\mathrm{N}}^2 > 0$. Otherwise we would have $\hat{\boldsymbol{r}}_{\mathrm{N}} = \hat{\boldsymbol{u}}_{\mathrm{N}} = 0$, which would mean that $\boldsymbol{z}_T = \mathbf{0}$ and either $\boldsymbol{z} = \mathbf{0}$ or $\boldsymbol{z} \in \operatorname{int}\mathscr{K}_1$.

B.1.2 Optimistic Newton algorithm

The algorithm below is used as part of the hybrid strategy to solve one-contact problems in the Gauss–Seidel algorithm from Chapter 4. As the initial guess from the enumerative solver is usually quite good, this algorithm needs not be extremely robust; for this reason, we only use a very slight damping. Note that this algorithm was still observed to perform quite well without the hybrid strategy, for instance for solving 6-dimensional problems from continuum granular simulations (Chapters 7 and 8).

Algorithm B.1: Optimistic Non-Smooth Newton algorithm

input : Initial guess $\boldsymbol{r}^0$
input : Matrix W, vector $\boldsymbol{b}$, friction coefficient μ
input : Curvature criterion $\sigma > 0$ (typically 10^{-2})
Result: $\boldsymbol{r}_{\text{best}}$ approximate solution to $f_{\text{FB}}^{\mathscr{K}}(\hat{\boldsymbol{r}}(\boldsymbol{r}), \hat{\boldsymbol{u}}(\boldsymbol{r})) = \boldsymbol{0}$
$\Phi_{\text{best}} \leftarrow +\infty$;
for $k = 0$ **to** *maxIters* **do**
  // Compute $f_{\text{FB}}^{\mathscr{K}}$ and its Jacobian at $\boldsymbol{r}^k$ (see Appendix B.1.1)
  $(f_{\text{FB}}, J_{\text{FB}}) \leftarrow$ computeFBandJacFB$(W, \boldsymbol{b}, \mu, \boldsymbol{r}^k)$;
  $\phi^k \leftarrow \frac{1}{2}\|f_{\text{FB}}\|^2$; // Compute value of error function
  if $\phi^k < \phi_{best}$ **then** // Check quality of current estimate
    $\boldsymbol{r}_{\text{best}} \leftarrow \boldsymbol{r}^k$;
    if $\phi^k < tol$ **then**
      break ; // Error low enough, exit algorithm
    end
  end
  $\nabla\Phi \leftarrow J_{\text{FB}}^{\mathsf{T}} f_{\text{FB}}$; // Compute gradient of objective function
  $\Delta\boldsymbol{r} \leftarrow -J_{\text{FB}}^{-1} f_{\text{FB}}$; // Compute new step $\Delta\boldsymbol{r}$
  if $\langle \Delta\boldsymbol{r}, \nabla\Phi \rangle > -\sigma\|\Delta\boldsymbol{r}\|\|\nabla\Phi\|$ **then** // Check for bad descent direction
    $\Delta\boldsymbol{r} \leftarrow \frac{1}{2}\Delta\boldsymbol{r}$;
  end
  $\boldsymbol{r}^{k+1} \leftarrow \boldsymbol{r}^k + \Delta\boldsymbol{r}$;
end

B.2 Suggested variants of the Projected Gradient Descent algorithm

The variant of the Projected Gradient Descent algorithm given below is implemented in the bogus library, and performed consistently well on our frictional contact and continuum problems (w.r.t. other algorithms of the same kind). However, theoretical convergence properties have not been studied.

Algorithm B.2 is a line-search free but Nesterov-accelerated variant of the Spectral Projected Gradient algorithm from Tasora (2013), and aims to derive an optimal descent step size from the previous iterates.

Procedure NesterovInertia(θ_{prev})
$\Delta \leftarrow \theta_{\text{prev}}\sqrt{\theta_{\text{prev}}^2 + 4}$;
$\theta \leftarrow \frac{1}{2}(\Delta - \theta_{\text{prev}}^2)$;
$\beta \leftarrow \frac{\theta_{\text{prev}}(1-\theta_{\text{prev}})}{\theta_{\text{prev}}^2+\theta}$;
return (θ, β) ;

Algorithm B.2: Accelerated Spectral Projected Gradient Descent

input : Initial guess $\boldsymbol{r}^0$
input : Matrix W, vector $\boldsymbol{b}$, orthogonal projectors Π_C
input : Min and max step sizes ξ_{min}, ξ_{max} (typically 10^{-6} and 10^{6})
Result: $\boldsymbol{r}_{\text{best}}$ approximate solution to $\min_{\boldsymbol{r}\in C} \frac{1}{2}\boldsymbol{r}^{\intercal} W\boldsymbol{r} + \boldsymbol{r}^T\boldsymbol{b}$

$\boldsymbol{r}^0 \leftarrow \Pi_C(\boldsymbol{r}^0)$;
$\boldsymbol{u}^0 \leftarrow W\boldsymbol{r}^0 + \boldsymbol{b}$; `// Gradient of objective function`
$\hat{\boldsymbol{r}}^0 \leftarrow \boldsymbol{r}^0$;
$\theta^k \leftarrow 1$; `// Inertia`
$\xi^0 \leftarrow$ `guessStepSize`(W)
$\Phi_{\text{best}} \leftarrow +\infty$;
for $k = 0$ **to** *maxIters* **do**
  $\phi^k \leftarrow$ `evaluateError`$(\boldsymbol{r}^k, \boldsymbol{u}^k)$; `// For instance using` $f_{\text{FB}}^{\mathscr{K}}$
  if $\phi^k < \phi_{best}$ **then**
    $\boldsymbol{r}_{\text{best}} \leftarrow \boldsymbol{r}^k$;
    if $\phi^k < tol$ **then break**; `// Error low enough, exit algorithm`
  end
B.2.13 $\boldsymbol{d}^k \leftarrow -\boldsymbol{u}^k$;
  $\hat{\boldsymbol{r}}^{k+1} \leftarrow \Pi_C\left(\boldsymbol{r}^k + \xi^k\boldsymbol{d}^k\right)$;
  if $\left\langle \hat{\boldsymbol{r}}^{k+1} - \hat{\boldsymbol{r}}^k, \boldsymbol{d}^k \right\rangle > 0$ **then** `// Check for descent direction`
    $(\theta^{k+1}, \beta) \leftarrow$ `NesterovInertia`(θ^k) ;
    $\boldsymbol{r}^{k+1} \leftarrow \hat{\boldsymbol{r}}^{k+1} + \beta\left(\hat{\boldsymbol{r}}^{k+1} - \hat{\boldsymbol{r}}^k\right)$;
  else
    $\boldsymbol{r}^{k+1} \leftarrow \hat{\boldsymbol{r}}^{k+1}$;
    $\theta^{k+1} \leftarrow 1$; `// Reset inertia`
  end
  $\boldsymbol{u}^{k+1} \leftarrow W\boldsymbol{r}^{k+1} + \boldsymbol{b}$; `// Gradient of objective function`
  $\xi^{k+1} \leftarrow \frac{\left\langle \boldsymbol{r}^{k+1} - \boldsymbol{r}^k, \boldsymbol{u}^{k+1} - \boldsymbol{u}^k \right\rangle}{\|\boldsymbol{u}^{k+1} - \boldsymbol{u}^k\|^2}$; `// Eq (5) from Barzilai and Borwein (1988)`
  $\xi^{k+1} \leftarrow \min(\xi_{max}, \max(\xi_{min}, \xi^{k+1}))$; `// Clamp next step size`
end

Algorithm B.2 can also be used to solve directly a DCFP by replacing line B.2.13 with
While this works relatively well in practice, the algorithm can no longer be interpreted as a

$\boldsymbol{d}^k \leftarrow -\boldsymbol{u}^k - \boldsymbol{s}(\boldsymbol{u}^k)$;

minimization algorithm.

B.3 Convergence of the out-of-order Gauss–Seidel algorithm

We study the convergence of the proximal, out-of-order Gauss–Seidel algorithm applied to the constrained minimization of a convex function outlined in Algorithm (B.3), where $\beta_i \geq 0 \forall 1 \leq i \leq n$ and the sequence $(j^k)_{k\in\mathbb{N}}$ denotes the index of the block that will be treated at the k^{th} iteration of the algorithm.

Algorithm 4.1 applied to the minimization of a SOCQP belongs to this class.

We make a few more hypothesis (which, again, are verified for Algorithm 4.1)

1. f is bounded from below;
2. f and ∇f are continuous;

Algorithm B.3: Out-of-order Proximal-Point algorithm
Input: $\boldsymbol{x}^0 \in C \subset X$
for $k \in \mathbb{N}$ **do**
$\boldsymbol{x}_{j^k}^{k+1} = \underset{\boldsymbol{y} \in C_{j^k}}{\arg\min}\, f_{j^k}(\boldsymbol{x}_1^k, \ldots, \boldsymbol{x}_{j^k-1}^k, \boldsymbol{y}, \boldsymbol{x}_{j^k+1}^k, \ldots, \boldsymbol{x}_n^k) + \frac{1}{2}\beta_{j^k}\|\boldsymbol{y} - \boldsymbol{x}_{j^k}^k\|^2$;
$\boldsymbol{x}^{k+1} = (\boldsymbol{x}_1^k, \ldots, \boldsymbol{x}_{j^k}^{k+1}, \ldots, \boldsymbol{x}_n^k)$
end

3. C is closed, convex and non-empty;
4. the maximum number of iterations between a local minimization for a given block and the next one for this same block is finite. Formally,

$$\exists M \in N \text{ s.t. } \forall 1 \leq i \leq n,\ \forall k \in \mathbb{N},\ \exists k^* \in (k, k+M) \text{ s.t. } j^{k^*} = i. \tag{B.1}$$

We will show that Algorithm B.3 converges to a solution of (B.2),

$$\underset{\boldsymbol{x} \in C}{\arg\min}\, f(\boldsymbol{x}), \tag{B.2}$$

as long as either f is strictly convex or $\beta_i > 0\ \forall i$.

The methodology below is adapted to our out-of-order setting from (Grippo and Sciandrone 2000).

Lemma B.1. *The sequence $(f(\boldsymbol{x}^k))_k$ generated by Algorithm B.3 converges to a limit $\bar{f}$.*

Proof. We first introduce the function $g^k : X \to R,\ \boldsymbol{z} \mapsto f(\boldsymbol{z}) + \frac{1}{2}\beta_{j^k}\|\boldsymbol{z} - \boldsymbol{x}^k\|^2$. By construction of Algorithm B.3,

$$\forall k \in \mathbb{N},\ f(\boldsymbol{x}^{k+1}) \leq g^k(\boldsymbol{x}^{k+1}) \leq g^k(\boldsymbol{x}^k) = f(\boldsymbol{x}^k).$$

The sequence $(f(\boldsymbol{x}^k))_k$ is thus decreasing and bounded from below, hence converges to a limit $\bar{f}$. □

Property B.1. *If for any $1 \leq i \leq n$, either $\beta_i > 0$ or f is strictly convex and coercive w.r.t. the i^{th} block, then the sequence $(\boldsymbol{x}^k)$ generated by Algorithm B.3 converges.*

In order to adapt the proof of Grippo and Sciandrone (2000), we first introduce a few supplemental notations.

Definition B.1. *Consider an infinite subset $K \subset \mathbb{N}$. For $m \in \mathbb{N}$, we define I_K^m as the set of all block indices that will be solved for infinitely many times at iteration $k+m$ for $k \in K$. Formally*

$$I_K^m := \left\{ i \in [1, n] \text{ s.t. } \forall k \in K, \exists k^* \in K, k^* \geq k \text{ and } j^{k^*+m} = i \right\}. \tag{B.3}$$

We also define for each block the subset $J_K^m(i) \subset K$ that contains all indices $k \in K$ such that the block i will be solved for at iteration $k+m$,

$$J_K^m(i) := \left\{ k \in K \text{ s.t. } j^{k+m} = i \right\}.$$

Note that the set $J_K^m(i)$ will be infinite if and only if $i \in I_K^m$.

Lemma B.2. *Let $\overline{J_K^m}$ denote the union of the sets $J_K^m(i)$ for $i \in I_K^m$,*

$$\overline{J_K^m} := \bigcup_{i \in I_K^m} J_K^m(i) \subset K.$$

For $m \in \mathbb{N}$, the subset $S_K^m := K \setminus \overline{J_K^m}$ is finite.

Proof. Suppose this was not the case, then the integer sequence $\left\{j^{k+m}\right\}_{k\in S_K^m} \subset [1,n]$ has a limit point i^* which is also in I_K^m as it satisfies Equation B.3. By definition of i^* there exists $k^* \in S_K^m$ such that $j^{k^*+m} = i^*$, which means $k^* \in J_K^m(i^*) \cap S_K^m = \emptyset$, which is a contradiction. □

We should now be able to prove Proposition B.1.

Proof. Let us denote by $B \subset [1,n]$ the set for which $\beta_i = 0$.

Existence of a limit point First, let us show that from $(\boldsymbol{x}^k)_{k\in\mathbb{N}}$ we can extract a converging subsequence. From Lemma B.1, we have $\lim_{k\in\mathbb{N}} f(\boldsymbol{x}^k) = \lim_{k\in\mathbb{N}} g^k(\boldsymbol{x}^{k+1}) = \bar{f}$, therefore $\forall 1 \leq i \leq n$, $\lim_{k\in\mathbb{N}} \beta_i \|\boldsymbol{x}^k - \boldsymbol{x}^{k+1}\| = 0$. Each block $\boldsymbol{x}_i^k$ for which $i \notin B$ thus admits a limit; let us define

$$\hat{\boldsymbol{x}}_i := \begin{cases} \lim \boldsymbol{x}_i^k & \text{if } i \notin B \\ \boldsymbol{0} & \text{if } i \in B. \end{cases}$$

The function $\hat{g} : \boldsymbol{z} \mapsto f(\boldsymbol{z}) + \frac{\beta_i}{2}\|\boldsymbol{z} - \hat{\boldsymbol{x}}\|^2$ is coercive on X and $\hat{g}(\boldsymbol{x}^k) \to \bar{f}$, therefore $(\boldsymbol{x}^k)_{k\in N}$ is bounded. Hence there exists a subsequence $(\boldsymbol{x}^k)_{k\in K}$ that converges to $\bar{\boldsymbol{x}} \in C$, and by continuity of f, $f(\bar{\boldsymbol{x}}) = \lim_{k\in\mathbb{N}} f(\boldsymbol{x}^K) = \bar{f}$. Necessarily, $\bar{\boldsymbol{x}}_i = \hat{\boldsymbol{x}}_i \ \forall i \notin B$.

Convergence of translated subsequences For $m \in \mathbb{N}$, we define the predicate $\Pi(m)$,

$$\Pi(m) := \lim_{k\in K} \boldsymbol{x}^{k+m} = \bar{\boldsymbol{x}}$$

$\Pi(0)$ hold by definition of K; now we suppose that $\Pi(m)$ is true, and we will show that this implies $\Pi(m+1)$.

For any $i \in I_k^{m+1} \cap B$, by definition the set $J_K^{m+1}(i) \subset K$ is infinite and thus $(\boldsymbol{x}^{k+m})_{k\in J_K^{m+1}(i)} \to \bar{\boldsymbol{x}}$. We can use (Grippo and Sciandrone 2000, Proposition 4) by identifying their $\{y^k\}$ with $(\boldsymbol{x}^{k+m})_{k\in J_K^{m+1}(i)}$ and their $\{v^k\}$ with $(\boldsymbol{x}^{k+m+1})_{k\in J_K^{m+1}(i)}$. We get

$$\lim_{k\in J_K^{m+1}(i)} \|\boldsymbol{x}_i^{k+m} - \boldsymbol{x}_i^{k+m+1}\| = 0$$

and therefore from $\Pi(m)$,

$$\lim_{k\in J_K^{m+1}(i)} \boldsymbol{x}_i^{k+m+1} = \bar{\boldsymbol{x}}_i. \tag{B.4}$$

Moreover, Equation (B.4) also holds for any $i \in I_k^{m+1} \setminus B$ since $(\boldsymbol{x}_i^k)_{k\in\mathbb{N}} \to \hat{\boldsymbol{x}}_i = \bar{\boldsymbol{x}}_i$. Now by definition of $J_K^{m+1}(i)$ and Algorithm B.3, $\forall i \in I_k^{m+1}$, $\forall \ell \neq i$, $\forall k \in J_K^{m+1}(i)$, $\boldsymbol{x}_\ell^{k+m} = \boldsymbol{x}_\ell^{k+m+1}$. Thus we get also from $\Pi(m)$ that $\forall i \in I_k^{m+1}$,

$$\lim_{k\in J_K^{m+1}(i)} \boldsymbol{x}_l^{k+m+1} = \lim_{k\in J_K^{m+1}(i)} \boldsymbol{x}_l^{k+m} = \bar{\boldsymbol{x}}_l \qquad \forall l \neq i. \tag{B.5}$$

Putting together (B.4) and (B.5), we get

$$\lim_{k\in J_K^{m+1}(i)} \boldsymbol{x}^{k+m+1} = \bar{\boldsymbol{x}}.$$

This being true for any $i \in I_k^{m+1}$, we deduce that

$$\lim_{k\in J_K^{m+1}} x^{k+m+1} = \bar{\boldsymbol{x}}$$

and with Lemma B.2,

$$\lim_{k\in K} x^{k+m+1} = \bar{\boldsymbol{x}}$$

which proves $\Pi(m+1)$. By induction $\Pi(m)$ holds for all $m \in \mathbb{N}$, which means that $(\boldsymbol{x}^k)_{k\in\mathbb{N}} \to \bar{\boldsymbol{x}}$. □

Property B.2. *Under the hypothesis of Property B.1, the limit $\bar{\boldsymbol{x}}$ of the sequence of iterates $(\boldsymbol{x}^k)$ from Algorithm B.3 is a critical point of the minimization problem (B.2).*

Again, let us first introduce an useful definition.

Definition B.2 (Successor function)**.** *Given Assumption B.1, we can define for each block $1 \leq i \leq n$ the function*

$$\mathrm{succ}_i : \begin{cases} \mathbb{N} \longrightarrow \mathbb{N} \\ k \longmapsto \min\left\{k^* \in (k, k+M), j^{k^*} = i\right\} \end{cases}$$

which indicates the next iteration at which a given block will be minimized over.

Proof. For any $1 \leq i \leq n$, the sequence $(\mathrm{succ}_i(k))_{k\in\mathbb{N}}$ is infinite and thus

$$\lim_{k\in\mathbb{N}} \boldsymbol{x}^{\mathrm{succ}_i(k)} = \bar{\boldsymbol{x}}. \tag{B.6}$$

Moreover by construction of Algorithm B.3, $(\nabla_i g^k)(\boldsymbol{x}^{\mathrm{succ}_i(k)}) \in -\mathcal{N}_{C_i}(\boldsymbol{x}_i^{\mathrm{succ}_i(k)})$, i.e.,

$$\left\langle (\nabla_i g^k)(\boldsymbol{x}^{\mathrm{succ}_i(k)}), \boldsymbol{z} - \boldsymbol{x}_i^{\mathrm{succ}_i(k)} \right\rangle \geq 0 \ \forall \boldsymbol{z} \in C_i.$$

This means

$$\left\langle (\nabla_i f^k)(\boldsymbol{x}^{\mathrm{succ}_i(k)}) + \beta_i(\boldsymbol{x}_i^{\mathrm{succ}_i(k)} - \boldsymbol{x}_i^{\mathrm{succ}_i(k)-1}), \boldsymbol{z} - \boldsymbol{x}_i^{\mathrm{succ}_i(k)} \right\rangle \geq 0 \quad \forall \boldsymbol{z} \in C_i$$

Taking into account Equation B.6 and the continuity assumption on $\nabla_i f$,

$$\langle (\nabla_i f)(\bar{\boldsymbol{x}}), \boldsymbol{y} - \bar{\boldsymbol{x}}_i \rangle \geq 0 \quad \forall \boldsymbol{y} \in C_i$$

i.e., $(\nabla_i f)(\bar{\boldsymbol{x}}) \in -\mathcal{N}_{C_i}(\bar{\boldsymbol{x}}_i)$. As this is true for any $1 \leq i \leq n$, $\bar{\boldsymbol{x}}$ is an optimal point of Problem B.2. □

C Supplemental justifications related to Drucker–Prager constraints

C.1 Constraints on quadrature points

The goal of this section is to justify Proposition 6.3, that Equations (6.13a — 6.13c) can be discretized as System (6.35).

Proof. First, it is noteworthy that the matrix M may be factorized as $M = R^T \operatorname{diag}(S) R$, where $\operatorname{diag}(S)$ is a diagonal matrix of size $\mathfrak{s}_d\, n_Q$ mapping to each quadrature point q the diagonal weight block $\mathfrak{S}_{q,q} := 2\, w_q\, I_{\mathfrak{s}_d}$. That is, the diagonal coefficients of S are given by

$$S_{(q-1)\mathfrak{s}_d+k} = 2\, w_q \quad \text{for } 1 \le k \le \mathfrak{s}_d \quad \text{and } 1 \le q \le n_Q.$$

Indeed, for all $1 \le i, j \le n$ and for all $1 \le k, \ell \le n$ let $r(i,k) := (i-1)\mathfrak{s}_d + k$ and $c(j,\ell) := (j-1)\mathfrak{s}_d + \ell$. Using the quadrature rule, we have

$$\begin{aligned}
M_{r,c} &= \sum_{q=1}^{n_Q} w_q \left(\mathbf{T}_{r(i,k)}(\hat{\boldsymbol{x}}_q) : \mathbf{T}_{c(j,\ell)}(\hat{\boldsymbol{x}}_q) \right) \\
&= 2 \sum_{q=1}^{n_Q} w_q \left\langle \omega_i^\tau(\hat{\boldsymbol{x}}_q) \chi(\boldsymbol{e}_k), \omega_j^\tau(\hat{\boldsymbol{x}}_q) \chi(\boldsymbol{e}_l) \right\rangle \\
&= 2 \sum_{q=1}^{n_Q} w_q\, \omega_i^\tau(\hat{\boldsymbol{x}}_q)\, \omega_j^\tau(\hat{\boldsymbol{x}}_q) \left(\boldsymbol{e}_k^\intercal \boldsymbol{e}_\ell \right) && \text{from (6.28)} \\
&= 2 \sum_{q=1}^{n_Q} w_q \left(\omega_i^\tau(\hat{\boldsymbol{x}}_q)\, \omega_j^\tau(\hat{\boldsymbol{x}}_q) \delta_{k,\ell} \right) \\
&= 2 \sum_{q=1}^{n_Q} \sum_{p=1}^{\mathfrak{s}_d} w_q \left(\omega_i^\tau(\hat{\boldsymbol{x}}_q)\, \omega_j^\tau(\hat{\boldsymbol{x}}_q) \delta_{k,p} \delta_{p,l} \right) \\
&= \sum_{q=1}^{n_Q} \sum_{p=1}^{\mathfrak{s}_d} S_{(q-1)\mathfrak{s}_d+p} \left(R_{(q-1)\mathfrak{s}_d+p,r(i,k)} R_{(q-1)\mathfrak{s}_d+p,c(j,\ell)} \right) && \text{from (6.34)},
\end{aligned}$$

which we recognize as the product $M = R^\intercal \operatorname{diag}(S) R$.

Furthermore, since M is invertible, we have $\operatorname{rank}(M) = \mathfrak{s}_d\, n$. From the rank inequality on product of matrices, we get

$$\operatorname{rank}(M) \le \min\left(\operatorname{rank}(\operatorname{diag}(S)), \operatorname{rank}(R)\right) \le \operatorname{rank}(R),$$

meaning that the rank of R is equal to its number of columns $\mathfrak{s}_d\, n$. The pseudoinverse $R^\dagger$ can therefore be computed as $R^\dagger = (R^\intercal R)^{-1} R^\intercal$, and $R^\dagger$ plays the role of a left inverse, i.e., we have $R^\dagger R = \mathbb{I}$.

We thus have $R^\dagger \hat{\underline{\boldsymbol{\lambda}}} = R^\dagger R\, \underline{\boldsymbol{\lambda}} = \underline{\boldsymbol{\lambda}}$, so (6.35a) is directly equivalent to (6.32a). Moreover, since for any strictly positive scalar ξ, $(\boldsymbol{\gamma}, \boldsymbol{\lambda}) \in \mathscr{DP}(\mu, \mathrm{Bi}, \zeta) \iff (\xi\boldsymbol{\gamma}, \boldsymbol{\lambda}) \in \mathscr{DP}(\mu, \mathrm{Bi}, \zeta)$, the

system is not affected by a multiplication of the left-hand-side of (6.35b) with the matrix $\mathrm{diag}(S)$. Multiplying then both sides of (6.35b) by $R^\intercal$, we get

$$R^\intercal \,\mathrm{diag}(S)\hat{\underline{\gamma}} = R^\intercal R^{\dagger,\intercal} B\,\underline{\boldsymbol{u}} + R^\intercal R^{\dagger,\intercal}\underline{\boldsymbol{k}}$$

Since $R^\intercal R^{\dagger,\intercal} = \left(R^\dagger R\right)^\intercal = \mathbb{I}$, we retrieve (6.32b).

Finally, as $\hat{\underline{\boldsymbol{\lambda}}}_{[q]} = \chi^{-1}(\boldsymbol{\lambda}(\hat{\boldsymbol{x}}_q))$ and $\hat{\underline{\boldsymbol{\gamma}}}_{[q]} = \chi^{-1}(\boldsymbol{\gamma}(\hat{\boldsymbol{x}}_q))$, we have (6.35c) $\iff \left(\boldsymbol{\gamma}(\hat{\boldsymbol{x}}_q), \boldsymbol{\lambda}(\hat{\boldsymbol{x}}_q)\right) \in \mathscr{DP}(\mu, \mathrm{Bi}, \zeta)$, that is, we ensure that $\forall \tau \in T_h$ the integral (6.33) is zero. □

C.2 Frictional boundaries

Suppose that $(\boldsymbol{\gamma}_{\mathrm{RB}}, \boldsymbol{\lambda}_{\mathrm{RB}}) \in \mathscr{DP}\mu_{\mathrm{RB}}$, with $\boldsymbol{\gamma}_{\mathrm{RB}} := \frac{1}{2}\left(\hat{\mathbf{v}}\boldsymbol{n}_{\mathrm{RB}}^\intercal + \boldsymbol{n}_{\mathrm{RB}}\hat{\mathbf{v}}^\intercal\right)$, and $\|\boldsymbol{n}_{\mathrm{RB}}\| = 1$.

The force induced by a stress $\boldsymbol{\sigma}$ through a plane with normal $\boldsymbol{n}$ is computed as $\boldsymbol{\sigma}\boldsymbol{n}$; the reaction force induced by the material on the frictional boundary is therefore $\boldsymbol{r} = \boldsymbol{\lambda}_{\mathrm{RB}}\boldsymbol{n}_{\mathrm{RB}}$. In the following, we investigate the relationship between $\boldsymbol{r}$ and $\hat{\mathbf{v}}$, the relative velocity of the boundary w.r.t. the granular material.

C.2.1 Signorini condition

First, remark that $\mathrm{Tr}\,\boldsymbol{\gamma}_{\mathrm{RB}} = \langle \hat{\mathbf{v}}, \boldsymbol{n}_{\mathrm{RB}} \rangle = \hat{\mathbf{v}}_{\mathrm{N}}$,

$$\boldsymbol{r}_{\mathrm{N}} = \frac{1}{3}(\mathrm{Tr}\,\boldsymbol{\lambda}_{\mathrm{RB}}) + (\mathrm{Dev}\,\boldsymbol{\lambda}_{\mathrm{RB}}) : (\boldsymbol{n}_{\mathrm{RB}}\boldsymbol{n}_{\mathrm{RB}}^\intercal)$$

and

$$\begin{aligned}(\mathrm{Dev}\,\boldsymbol{\lambda}_{\mathrm{RB}}) : (\boldsymbol{n}_{\mathrm{RB}}\boldsymbol{n}_{\mathrm{RB}}^\intercal) &\le 2\,|\,\mathrm{Dev}\,\boldsymbol{\lambda}_{\mathrm{RB}}||\boldsymbol{n}_{\mathrm{RB}}\boldsymbol{n}_{\mathrm{RB}}^\intercal| \\ &\le \frac{\mu_{\mathrm{RB}}}{\sqrt{6}}\,\mathrm{Tr}\,\boldsymbol{\lambda}_{\mathrm{RB}} \times \frac{2}{\sqrt{2}} \\ &\le \frac{\mu_{\mathrm{RB}}}{\sqrt{3}}\,\mathrm{Tr}\,\boldsymbol{\lambda}_{\mathrm{RB}}\,.\end{aligned}$$

Therefore,

$$\left(\frac{1-\sqrt{3}\mu_{\mathrm{RB}}}{3}\right)\mathrm{Tr}\,\boldsymbol{\lambda}_{\mathrm{RB}} \le \boldsymbol{r}_{\mathrm{N}} \le \left(\frac{1+\sqrt{3}\mu_{\mathrm{RB}}}{3}\right)\mathrm{Tr}\,\boldsymbol{\lambda}_{\mathrm{RB}}\,.$$

For $\mu_{\mathrm{RB}} < \frac{1}{\sqrt{3}}$, we thus have

$$0 \le \mathrm{Tr}\,\boldsymbol{\gamma}_{\mathrm{RB}} \perp \mathrm{Tr}\,\boldsymbol{\lambda}_{\mathrm{RB}} \ge 0 \implies 0 \le \hat{\mathbf{v}}_{\mathrm{N}} \perp \boldsymbol{r}_{\mathrm{N}} \ge 0\,,$$

i.e., the Signorini condition is satisfied.

C.2.2 Tangential reaction

Notice that

$$\begin{aligned}\mathrm{Dev}(\boldsymbol{\gamma}_{\mathrm{RB}})\boldsymbol{n}_{\mathrm{RB}} &= \frac{1}{2}\hat{\mathbf{v}} + (\frac{1}{2} - \frac{1}{3})\,\langle \hat{\mathbf{v}}, \boldsymbol{n}_{\mathrm{RB}} \rangle\,\boldsymbol{n}_{\mathrm{RB}} \\ &= \frac{1}{2}\hat{\mathbf{v}}_{\mathbf{T}} + \frac{2}{3}\hat{\mathbf{v}}_{\mathrm{N}}\boldsymbol{n}\,.\end{aligned}$$

Sliding case First suppose that $\hat{\mathbf{v}}_{\mathbf{T}} \neq 0$, therefore $\mathrm{Dev}\,\boldsymbol{\gamma}_{\mathrm{RB}} \neq 0$, and $\mathscr{DP}\mu_{\mathrm{RB}}$ imposes that $\mathrm{Dev}\,\boldsymbol{\lambda}_{\mathrm{RB}} = -\alpha\,\mathrm{Dev}\,\boldsymbol{\gamma}_{\mathrm{RB}}, \alpha > 0$. Since $\boldsymbol{r} = \frac{1}{3}\mathrm{Tr}\,\boldsymbol{\lambda}_{\mathrm{RB}}\boldsymbol{n} + \mathrm{Dev}\,\boldsymbol{\lambda}_{\mathrm{RB}}\boldsymbol{n}$, we can identify that

$$\boldsymbol{r}_{\mathbf{T}} = -\frac{1}{2}\alpha\hat{\mathbf{v}}_{\mathbf{T}}$$

i.e. the tangential friction force is opposed to the tangential relative velocity. Now, let us show that $\boldsymbol{r}$ lies on the boundary of the second-order cone of aperture $\sqrt{\frac{3}{2}}\mu_{\mathrm{RB}}$, i.e., $\|\boldsymbol{r}_{\mathbf{T}}\| = \sqrt{\frac{3}{2}}\mu_{\mathrm{RB}}\boldsymbol{r}_{\mathrm{N}}$.

From the Signori condition, $\hat{\mathbf{v}}_{\mathrm{N}} > 0$ implies $\boldsymbol{r}_{\mathrm{N}} = 0$, and therefore $\operatorname{Tr}\boldsymbol{\lambda}_{\mathrm{RB}} = 0$. This means $|\operatorname{Dev}\boldsymbol{\lambda}_{\mathrm{RB}}| = 0$, and consequently $\|\boldsymbol{r}_{\mathbf{T}}\|$. Our relation $\|\boldsymbol{r}_{\mathbf{T}}\| = \sqrt{\frac{3}{2}}\mu_{\mathrm{RB}}\boldsymbol{r}_{\mathrm{N}}$ is trivially satisfied.

We now have to study the case $\hat{\mathbf{v}}_{\mathrm{N}} = 0$. $\operatorname{Dev}\boldsymbol{\lambda}_{\mathrm{RB}} = -\alpha \operatorname{Dev}\boldsymbol{\gamma}_{\mathrm{RB}}$ means that

$$\|\boldsymbol{r}_{\mathbf{T}}\| = \|\operatorname{Dev}\boldsymbol{\lambda}_{\mathrm{RB}}\boldsymbol{n}\| = |\operatorname{Dev}\boldsymbol{\lambda}_{\mathrm{RB}}\boldsymbol{n}|\frac{\|\operatorname{Dev}\boldsymbol{\gamma}_{\mathrm{RB}}\boldsymbol{n}_{\mathrm{RB}}\|}{|\operatorname{Dev}\boldsymbol{\gamma}_{\mathrm{RB}}|}.$$

Since $\hat{\mathbf{v}}.\boldsymbol{n}_{\mathrm{RB}} = 0$,

$$\|\operatorname{Dev}(\boldsymbol{\gamma}_{\mathrm{RB}})\boldsymbol{n}_{\mathrm{RB}}\| = \frac{1}{2}\|\hat{\mathbf{v}}_{\mathbf{T}}\|$$

and

$$\begin{aligned}|\operatorname{Dev}(\boldsymbol{\gamma}_{\mathrm{RB}})|^2 = |\boldsymbol{\gamma}_{\mathrm{RB}}|^2 &= |\hat{\mathbf{v}}\boldsymbol{n}_{\mathrm{RB}}^{\intercal}|^2 - \frac{1}{4}|\hat{\mathbf{v}}\boldsymbol{n}_{\mathrm{RB}}^{\intercal} - \boldsymbol{n}_{\mathrm{RB}}\hat{\mathbf{v}}^{\intercal}|^2\\ &= \frac{1}{2}\|\hat{\mathbf{v}}\|^2 - \frac{1}{4}\|\hat{\mathbf{v}}\wedge\boldsymbol{n}_{\mathrm{RB}}\|^2 = \frac{1}{2}\|\hat{\mathbf{v}}\|^2 - \frac{1}{4}\|\hat{\mathbf{v}}_{\mathbf{T}}\|^2\\ &= \frac{1}{4}\|\hat{\mathbf{v}}_{\mathbf{T}}\|^2 = \|\operatorname{Dev}(\boldsymbol{\gamma}_{\mathrm{RB}})\boldsymbol{n}_{\mathrm{RB}}\|^2.\end{aligned}$$

This means

$$\|\boldsymbol{r}_{\mathbf{T}}\| = |\operatorname{Dev}\boldsymbol{\lambda}_{\mathrm{RB}}| = \frac{\mu_{\mathrm{RB}}}{\sqrt{6}}\operatorname{Tr}\boldsymbol{\lambda}_{\mathrm{RB}} = \sqrt{\frac{3}{2}}\mu_{\mathrm{RB}}\boldsymbol{r}_{\mathrm{N}}.$$

The sliding case thus satisfies the Coulomb law with coefficient $\sqrt{\frac{3}{2}}\mu_{\mathrm{RB}}$.

Sticking When $\hat{\mathbf{v}} = 0$, we cannot conclude without more information about the relationship between $\boldsymbol{\lambda}_{\mathrm{RB}}$ and $\boldsymbol{\gamma}_{\mathrm{RB}}$. Indeed, we can only verify that

$$\begin{aligned}\|\boldsymbol{r}_{\mathbf{T}}\| &\leq \|\operatorname{Dev}\boldsymbol{\lambda}_{\mathrm{RB}}\boldsymbol{n}\| \leq \sqrt{2}|\operatorname{Dev}\boldsymbol{\lambda}_{\mathrm{RB}}|\\ &\leq \frac{\mu_{\mathrm{RB}}}{\sqrt{3}}\operatorname{Tr}\boldsymbol{\lambda}_{\mathrm{RB}} \leq \frac{\sqrt{3}\mu_{\mathrm{RB}}}{1-\sqrt{3}\mu_{\mathrm{RB}}}\boldsymbol{r}_{\mathrm{N}}\end{aligned}$$

i.e the reaction force has to lie inside a second-order cone of aperture $\frac{\sqrt{3}\mu_{\mathrm{RB}}}{1-\sqrt{3}\mu_{\mathrm{RB}}}$.

This last bound does not correspond to the one derived for the sliding case (except when $\mu_{\mathrm{RB}} = 0$), but nevertheless models a coupling between the tangential and normal reaction forces.

C.2.3 Reverse inclusion

For any $(\hat{\mathbf{v}}, \boldsymbol{r}) \in \mathscr{C}_{\sqrt{\frac{3}{2}}\mu}(\boldsymbol{n}_{\mathrm{RB}})$ — i.e., satisfying the 3D Coulomb law with friction coefficient $\sqrt{\frac{3}{2}}\mu$ — we can construct a symmetric tensor $\boldsymbol{\lambda}_{\mathrm{RB}}$ such that $(\boldsymbol{\gamma}_{\mathrm{RB}}, \boldsymbol{\lambda}_{\mathrm{RB}}) \in \mathscr{DP}(\mu)$. Indeed, let

$$\boldsymbol{\lambda}_{\mathrm{RB}} := \left(\boldsymbol{r}_{\mathbf{T}}\boldsymbol{n}_{\mathrm{RB}}^{\intercal} + \boldsymbol{n}_{\mathrm{RB}}\boldsymbol{r}_{\mathbf{T}}^{\intercal}\right) + \boldsymbol{r}_{\mathrm{N}}\mathbb{I}$$

We have

$$\begin{aligned}\operatorname{Tr}\boldsymbol{\lambda}_{\mathrm{RB}} &= 3\boldsymbol{r}_{\mathrm{N}} = \sqrt{6}\sqrt{\frac{3}{2}}\boldsymbol{r}_{\mathrm{N}}\\ \boldsymbol{\lambda}_{\mathrm{RB}}\boldsymbol{n}_{\mathrm{RB}} &= \boldsymbol{r}_{\mathbf{T}} + \boldsymbol{n}_{\mathrm{RB}}(\boldsymbol{r}_{\mathrm{N}}) = \boldsymbol{r}\\ \operatorname{Dev}(\boldsymbol{\lambda}_{\mathrm{RB}})\boldsymbol{n}_{\mathrm{RB}} &= \boldsymbol{r}_{\mathbf{T}}\\ |\operatorname{Dev}(\boldsymbol{\lambda}_{\mathrm{RB}})| &= \left|\boldsymbol{r}_{\mathbf{T}}\boldsymbol{n}_{\mathrm{RB}}^{\intercal} + \boldsymbol{n}_{\mathrm{RB}}(\boldsymbol{r} - \boldsymbol{r}_{\mathrm{N}}\boldsymbol{n}_{\mathrm{RB}})^{\intercal}\right|\\ &= \|\boldsymbol{r}_{\mathbf{T}}\|\end{aligned}$$

It can be easily verified that for any case of the $\mathscr{C}_{\sqrt{\frac{3}{2}}\mu}(\boldsymbol{n}_{\mathrm{RB}})$ disjunctive formulation satisfied by $\boldsymbol{r}$ and $\hat{\mathbf{v}}$, the corresponding case of $\mathscr{DP}(\mu)$ is satisfied by $(\boldsymbol{\gamma}_{\mathrm{RB}}, \boldsymbol{\lambda}_{\mathrm{RB}})$.

Bibliography

Acary, V. and B. Brogliato (2008). *Numerical methods for nonsmooth dynamical systems*. English. Vol. 35. Lecture Notes in Computational and Applied Mechanics. Springer.

Acary, V. (2009). *Toward higher order event–capturing schemes and adaptive time–step strategies for nonsmooth multibody systems*. Research Report RR-7151, p. 31. URL: https://hal.inria.fr/inria-00440771.

Acary, V., M. Brémond, T. Koziara, and F. Pérignon (2014). *FCLIB: a collection of discrete 3D Frictional Contact problems*. Technical Report RT-0444. INRIA, p. 34. URL: https://hal.inria.fr/hal-00945820.

Ainsley, S., E. Vouga, E. Grinspun, and R. Tamstorf (2012). "Speculative Parallel Asynchronous Contact Mechanics". In: *ACM Trans. Graph.* 31.6, 151:1–151:8. ISSN: 0730-0301. DOI: 10.1145/2366145.2366170.

Alart, P. and A. Curnier (1991). "A mixed formulation for frictional contact problems prone to Newton like solution methods". In: *Computer Methods in Applied Mechanics and Engineering* 92.3, pp. 353–375. ISSN: 0045-7825. DOI: 10.1016/0045-7825(91)90022-x.

Alduán, I. and M. A. Otaduy (2011). "SPH Granular Flow with Friction and Cohesion". In: *Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. SCA '11. Vancouver, British Columbia, Canada: ACM, pp. 25–32. ISBN: 978-1-4503-0923-3. DOI: 10.1145/2019406.2019410.

Alduán, I., A. Tena, and M. A. Otaduy (2009). "Simulation of high-resolution granular media". In: *Proc. of Congreso Español de Informática Gráfica*. Vol. 1. 4.

Alejano, L. R. and A. Bobet (2012). "Drucker–Prager Criterion". In: *Rock Mechanics and Rock Engineering* 45.6, pp. 995–999. ISSN: 1434-453X. DOI: 10.1007/s00603-012-0278-2.

Andersen, E. and K. Andersen (2013). *MOSEK modeling manual*.

Andersen, E., C. Roos, and T. Terlaky (2003). "On implementing a primal-dual interior-point method for conic quadratic optimization". In: *Mathematical Programming* 95.2, pp. 249–277. ISSN: 1436-4646. DOI: 10.1007/s10107-002-0349-3.

Andersen, S. and L. Andersen (2010). "Analysis of spatial interpolation in the material-point method". In: *Computers & Structures* 88.7-8, pp. 506–518. ISSN: 0045-7949. DOI: 10.1016/j.compstruc.2010.01.004.

Anderson, T. B. and R. Jackson (1967). "Fluid Mechanical Description of Fluidized Beds. Equations of Motion". In: *Industrial & Engineering Chemistry Fundamentals* 6.4, pp. 527–539. ISSN: 1541-4833. DOI: 10.1021/i160024a007.

Andreotti, B., Y. Forterre, and O. Pouliquen (2011). *Granular media: between fluid and solid*. Cambridge University Press.

Andrews, M. and P. O'Rourke (1996). "The multiphase particle-in-cell (MP-PIC) method for dense particulate flows". In: *International Journal of Multiphase Flow* 22.2, pp. 379–402. ISSN: 0301-9322. DOI: 10.1016/0301-9322(95)00072-0.

Anitescu, M. (2005). "Optimization-based simulation of nonsmooth rigid multibody dynamics". In: *Math. Program.* 105.1, pp. 113–143. ISSN: 1436-4646. DOI: 10.1007/s10107-005-0590-7.

Anjyo, K., Y. Usami, and T. Kurihara (1992). "A Simple Method for Extracting the Natural Beauty of Hair". In: *Computer Graphics Proceedings (Proceedings of the ACM SIGGRAPH'92 conference)*, pp. 111–120.

Apte, S., K. Mahesh, and T. Lundgren (2003). "An Eulerian-Lagrangian Model to Simulate Two-Phase/Particulate Flows in Complex Geometries". In: *APS Division of Fluid Dynamics Meeting Abstracts*, A1.

Aubry, J.-M. and X. Xian (2015). "Fast Implicit Simulation of Flexible Trees". In: *Mathematics for Industry*, pp. 47–61. ISSN: 2198-3518. DOI: 10.1007/978-4-431-55483-7_5.

Autodesk (2009). *3ds Max Hair & Fur*. URL: http://usa.autodesk.com/.

Barbu, V. and T. Precupanu (2012). "Convex Functions". In: *Convexity and Optimization in Banach Spaces*, pp. 67–151. ISSN: 1439-7382. DOI: 10.1007/978-94-007-2247-7_2.

Bardenhagen, S. G. and E. M. Kober (2004). "The Generalized Interpolation Material Point Method". In: *CMES: Computer Modeling in Engineering & Sciences* 5.6, pp. 477–496. DOI: 10.3970/cmes.2004.005.477.

Bardenhagen, S., J. Brackbill, and D. Sulsky (2000). "The material-point method for granular materials". In: *Computer Methods in Applied Mechanics and Engineering* 187.3-4, pp. 529–541. ISSN: 0045-7825. DOI: 10.1016/s0045-7825(99)00338-2.

Barker, T., D. G. Schaeffer, P. Bohorquez, and J. M. N. T. Gray (2015). "Well-posed and ill-posed behaviour of the -rheology for granular flow". In: *Journal of Fluid Mechanics* 779, pp. 794–818. ISSN: 1469-7645. DOI: 10.1017/jfm.2015.412.

Barzilai, J. and J. M. Borwein (1988). "Two-Point Step Size Gradient Methods". In: *IMA Journal of Numerical Analysis* 8.1, pp. 141–148. ISSN: 1464-3642. DOI: 10.1093/imanum/8.1.141.

Batchelor, G. K. (1988). "A new theory of the instability of a uniform fluidized bed". In: *Journal of Fluid Mechanics* 193.-1, p. 75. ISSN: 1469-7645. DOI: 10.1017/s002211208800206x.

Batty, C., F. Bertails, and R. Bridson (2007). "A fast variational framework for accurate solid-fluid coupling". In: *ACM Transactions on Graphics* 26.3, p. 100. ISSN: 0730-0301. DOI: 10.1145/1276377.1276502.

Bedford, A. and D. Drumheller (1983). "Theories of immiscible and structured mixtures". In: *International Journal of Engineering Science* 21.8, pp. 863–960. ISSN: 0020-7225. DOI: 10.1016/0020-7225(83)90071-x.

Berga, A. and G. de Saxcé (1994). "Elastoplastic Finite Element Analysis of Soil Problems with Implicit Standard Material Constitutive Laws". In: *Revue Européenne des Éléments* 3.3, pp. 411–456. DOI: 10.1080/12506559.1994.10511137. eprint: http://www.tandfonline.com/doi/pdf/10.1080/12506559.1994.10511137.

Bergou, M., B. Audoly, E. Vouga, M. Wardetzky, and E. Grinspun (2010). "Discrete Viscous Threads". In: *ACM Transactions on Graphics (Proceedings of the ACM SIGGRAPH'10 Conference)*.

Bergou, M., M. Wardetzky, S. Robinson, B. Audoly, and E. Grinspun (2008). "Discrete elastic rods". In: *ACM Transactions on Graphics (Proceedings of the ACM SIGGRAPH'08 conference)* 27.3, pp. 1–12. ISSN: 0730-0301. DOI: http://doi.acm.org/10.1145/1360612.1360662.

Bertails, F., B. Audoly, M.-P. Cani, B. Querleux, et al. (2006). "Super-helices for predicting the dynamics of natural hair". In: *ACM Transactions on Graphics – SIGGRAPH 2006*. DOI: 10.1145/1179352.1142012.

Bertails-Descoubes, F., F. Cadoux, G. Daviet, and V. Acary (2011). "A nonsmooth Newton solver for capturing exact Coulomb friction in fiber assemblies". In: *ACM Transactions on Graphics* 30 (1), 6:1–6:14. ISSN: 0730-0301. DOI: http://doi.acm.org/10.1145/1899404.1899410.

Besson, U. (2007). "Du frottement à la tribologie: survol historique". In: *Bulletin de l'Union des physiciens* 899, pp. 1137–1154.

Beverloo, W., H. Leniger, and J. van de Velde (1961). "The flow of granular solids through orifices". In: *Chemical Engineering Science* 15.3-4, pp. 260–269. ISSN: 0009-2509. DOI: 10.1016/0009-2509(61)85030-6.

Biot, M. A. (1955). "Theory of Elasticity and Consolidation for a Porous Anisotropic Solid". In: *Journal of Applied Physics* 26.2, p. 182. ISSN: 0021-8979. DOI: 10.1063/1.1721956.

Birman, V. K., J. E. Martin, and E. Meiburg (2005). "The non-Boussinesq lock-exchange problem. Part 2. High-resolution simulations". In: *Journal of Fluid Mechanics* 537.-1, p. 125. ISSN: 1469-7645. DOI: 10.1017/s0022112005005033.

Bleyer, J., M. Maillard, P. de Buhan, and P. Coussot (2015). "Efficient numerical computations of yield stress fluid flows using second-order cone programming". In: *Computer Methods in Applied Mechanics and Engineering* 283, pp. 599–614. ISSN: 0045-7825. DOI: 10.1016/j.cma.2014.10.008.

Bonnefon, O. and G. Daviet (2011). *Quartic formulation of Coulomb 3D frictional contact*. Anglais. Tech. rep. INRIA - Laboratoire Jean Kuntzmann. URL: http://hal.archives-ouvertes.fr/inria-00553859/en/.

Borwein, J. M. and D. R. Luke (2011). "Duality and convex programming". In: *Handbook of Mathematical Methods in Imaging*. Springer, pp. 229–270.

Bouchut, F., R. Eymard, and A. Prignet (2014). "Convergence of conforming approximations for inviscid incompressible Bingham fluid flows and related problems". In: *Journal of Evolution Equations* 14.3, pp. 635–669. ISSN: 1424-3202. DOI: 10.1007/s00028-014-0231-9.

Bouchut, F., E. D. Fernández-Nieto, A. Mangeney, and G. Narbona-Reina (2016). "A two-phase two-layer model for fluidized granular flows with dilatancy effects". In: *Journal of Fluid Mechanics* 801, pp. 166–221. ISSN: 1469-7645. DOI: 10.1017/jfm.2016.417.

Bouchut, F., I. R. Ionescu, and A. Mangeney (2016). "An analytic approach for the evolution of the static/flowing interface in viscoplastic granular flows". working paper or preprint. URL: https://hal-upec-upem.archives-ouvertes.fr/hal-01081213.

Bowden, F. P. and D. Tabor (1950). *The friction and lubrication of solids*. Vol. 1. Oxford university press.

Boyd, L. and R. Bridson (2012). "MultiFLIP for energetic two-phase fluid simulation". In: *ACM Transactions on Graphics* 31.2, pp. 1–12. ISSN: 0730-0301. DOI: 10.1145/2159516.2159522.

Boyer, F. (2001). "Ecoulements diphasiques de type Cahn-Hilliard". Thèse de doctorat dirigée par Fabrie, Pierre Mathématiques appliquées Bordeaux 1 2001. PhD thesis. Université Bordeaux 1, 191 p. URL: http://www.theses.fr/2001BOR10509.

Boyer, F., É. Guazzelli, and O. Pouliquen (2011). "Unifying Suspension and Granular Rheology". In: *Physical Review Letters* 107.18. ISSN: 1079-7114. DOI: 10.1103/physrevlett.107.188301.

Brackbill, J. and H. Ruppel (1986). "FLIP: A method for adaptively zoned, particle-in-cell calculations of fluid flows in two dimensions". In: *Journal of Computational Physics* 65.2, pp. 314–343. ISSN: 0021-9991. DOI: 10.1016/0021-9991(86)90211-1.

Bridson, R., R. Fedkiw, and J. Anderson (2002). "Robust treatment of collisions, contact and friction for cloth animation". In: *Proceedings of SIGGRAPH '02*. ISSN: 0730-0301. DOI: 10.1145/566570.566623.

Brochu, T., E. Edwards, and R. Bridson (2012). "Efficient geometrically exact continuous collision detection". In: *ACM Transactions on Graphics (TOG)* 31.4, p. 96.

Brogliato, B. (1999). *Nonsmooth Mechanics: Models, Dynamics and Control (Communications and Control Engineering)*. Springer. ISBN: 1447111613.

Bürger, R. (2000). "Phenomenological foundation and mathematical theory of sedimentation–consolidation processes". In: *Chemical Engineering Journal* 80.1, pp. 177–188.

Burgess, D., D. Sulsky, and J. Brackbill (1992). "Mass matrix formulation of the FLIP particle-in-cell method". In: *Journal of Computational Physics* 103.1, pp. 1–15. ISSN: 0021-9991. DOI: 10.1016/0021-9991(92)90323-q.

Burmeister, W. (1985). "Private communication, Dresden".

Cadoux, F. (2009). "Optimization and convex analysis for nonsmooth dynamics". Theses. Université Joseph-Fourier - Grenoble I. URL: https://tel.archives-ouvertes.fr/tel-00440798.

Casati, R. (2015). "Quelques contributions à la modélisation numérique de structures élancées pour l'informatique graphique". PhD thesis. Université Grenoble Alpes.

Casati, R., G. Daviet, and F. Bertails-Descoubes (2016). *Inverse Elastic Cloth Design with Contact and Friction*. Research Report. Inria Grenoble Rhône-Alpes, Université de Grenoble. URL: https://hal.archives-ouvertes.fr/hal-01309617.

Chambolle, A. and T. Pock (2010). "A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging". In: *J Math Imaging Vis* 40.1, pp. 120–145. ISSN: 1573-7683. DOI: 10.1007/s10851-010-0251-1.

Chambon, G., R. Bouvarel, D. Laigle, and M. Naaim (2011). "Numerical simulations of granular free-surface flows using smoothed particle hydrodynamics". In: *Journal of Non-Newtonian Fluid Mechanics* 166.12-13, pp. 698–712. ISSN: 0377-0257. DOI: 10.1016/j.jnnfm.2011.03.007.

Chauchat, J., S. Guillou, D. Pham Van Bang, and K. Dan Nguyen (2013). "Modelling sedimentation-consolidation in the framework of a one-dimensional two-phase flow model". In: *Journal of Hydraulic Research* 51.3, pp. 293–305. ISSN: 1814-2079. DOI: 10.1080/00221686.2013.768798.

Chauchat, J. and M. Médale (2010). "A three-dimensional numerical model for incompressible two-phase flow of a granular bed submitted to a laminar shearing flow". In: *Computer Methods in Applied Mechanics and Engineering* 199.9-12, pp. 439–449. ISSN: 0045-7825. DOI: 10.1016/j.cma.2009.07.007.

— (2014). "A three-dimensional numerical model for dense granular flows based on the mu(I) rheology". In: *Journal of Computational Physics* 256, pp. 696–712. ISSN: 0021-9991. DOI: 10.1016/j.jcp.2013.09.004.

Chehata, D., R. Zenit, and C. R. Wassgren (2003). "Dense granular flow around an immersed cylinder". In: *Physics of Fluids* 15.6, p. 1622. ISSN: 1070-6631. DOI: 10.1063/1.1571826.

Chen, J.-S. and P. Tseng (2005). "An unconstrained smooth minimization reformulation of the second-order cone complementarity problem". In: *Math. Program.* 104 (2), pp. 293–327. ISSN: 0025-5610. DOI: 10.1007/s10107-005-0617-0.

Choe, B., M. Choi, and H.-S. Ko (2005). "Simulating complex hair with robust collision handling". In: *Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*. SCA '05. Los Angeles, California: ACM, pp. 153–160. ISBN: 1-59593-198-8. DOI: http://doi.acm.org/10.1145/1073368.1073389.

Choi, H. G. and D. D. Joseph (2001). "Fluidization by lift of 300 circular particles in plane Poiseuille flow by direct numerical simulation". In: *Journal of Fluid Mechanics* 438. ISSN: 1469-7645. DOI: 10.1017/s0022112001004177.

Concha, F. and R. Bürger (2002). "A century of research in sedimentation and thickening". In: *Kona* 20, pp. 38–70.

Coulomb, C.-A. (1781). *Théorie des machines simples: en ayant égard au frottement de leurs parties et à la roideur des cordages*.

Cundall, P. A. and O. D. L. Strack (1979). "A discrete numerical model for granular assemblies". In: *Géotechnique* 29.1, pp. 47–65. ISSN: 1751-7656. DOI: 10.1680/geot.1979.29.1.47.

Dahl, P. R. (1968). *A solid friction model*. Tech. rep. The Aerospace Corporation.

Daviet, G. (2013). *So-bogus, a C++ linear algebra and solvers library for sparse block matrices*. http://gdaviet.fr/code/bogus.

Daviet, G. and F. Bertails-Descoubes (2016a). "A Semi-Implicit Material Point Method for the Continuum Simulation of Granular Materials". In: *ACM Transactions on Graphics*. SIGGRAPH '16 Technical Papers 35.4, p. 13. DOI: 10.1145/2897824.2925877.

— (2016b). "Nonsmooth simulation of dense granular flows with pressure-dependent yield stress". In: *Journal of Non-Newtonian Fluid Mechanics* 234, pp. 15–35. ISSN: 0377-0257. DOI: http://dx.doi.org/10.1016/j.jnnfm.2016.04.006.

Daviet, G., F. Bertails-Descoubes, and L. Boissieux (2011). "A hybrid iterative solver for robustly capturing Coulomb friction in hair dynamics". In: *ACM Transactions on Graphics* 30.6, pp. 1–12. ISSN: 0730-0301. DOI: 10.1145/2070781.2024173.

Daviet, G., F. Bertails-Descoubes, and R. Casati (2015). "Fast cloth simulation with implicit contact and exact coulomb friction". In: *Proceedings of the 14th ACM SIGGRAPH / Eurographics Symposium on Computer Animation - SCA '15*. DOI: 10.1145/2786784.2795139.

De Saxcé, G. (1992). “Une généralisation de l’inégalité de Fenchel et ses applications aux lois constitutives”. In: *Comptes rendus de l’Académie des sciences. Série 2, Mécanique, Physique, Chimie, Sciences de l’univers, Sciences de la Terre* 314.2, pp. 125–129.

De Saxcé, G. and Z.-Q. Feng (1991). “New Inequality and Functional for Contact with Friction: The Implicit Standard Material Approach”. In: *Mechanics of Structures and Machines* 19.3, pp. 301–325. DOI: 10.1080/08905459108905146. eprint: http://dx.doi.org/10.1080/08905459108905146.

— (1998). “The bipotential method: A constructive approach to design the complete contact law with friction and improved numerical algorithms”. In: *Mathematical and Computer Modelling* 28.4-8, pp. 225–245. ISSN: 0895-7177. DOI: 10.1016/s0895-7177(98)00119-8.

De Saxcé, G. and L. Bousshine (2002). “Implicit Standard Materials”. In: *Inelastic Behaviour of Structures under Variable Repeated Loads*, pp. 59–76. DOI: 10.1007/978-3-7091-2558-8_4.

Denlinger, R. P. and R. M. Iverson (2001). “Flow of variably fluidized granular masses across three-dimensional terrain: 2. Numerical predictions and experimental tests”. In: *Journal of Geophysical Research* 106.B1, p. 553. ISSN: 0148-0227. DOI: 10.1029/2000jb900330.

Derouet-Jourdan, A. (2013). “Dynamic curves : from geometrical shape capture to deformable objects animation.” Theses. Université de Grenoble. URL: https://tel.archives-ouvertes.fr/tel-01135185.

Derouet-Jourdan, A., F. Bertails-Descoubes, G. Daviet, and J. Thollot (2013). “Inverse dynamic hair modeling with frictional contact”. In: *ACM Transactions on Graphics* 32.6, pp. 1–10. ISSN: 0730-0301. DOI: 10.1145/2508363.2508398.

Derouet-Jourdan, A., F. Bertails-Descoubes, and J. Thollot (2010). “Stable inverse dynamic curves”. In: *ACM Transactions on Graphics (Proceedings of the ACM SIGGRAPH Asia’10 conference)*. DOI: 10.1145/1882262.1866159.

— (2013). “Floating tangents for approximating spatial curves with G^1 piecewise helices”. In: *Computer Aided Geometric Design* 30.5, pp. 490–520. ISSN: 0167-8396. DOI: 10.1016/j.cagd.2013.02.007.

Domnik, B. and S. P. Pudasaini (2012). “Full two-dimensional rapid chute flows of simple viscoplastic granular materials with a pressure-dependent dynamic slip-velocity and their numerical simulations”. In: *Journal of Non-Newtonian Fluid Mechanics* 173-174, pp. 72–86. ISSN: 0377-0257. DOI: 10.1016/j.jnnfm.2012.03.001.

Drew, D. A. (1983). “Mathematical Modeling of Two-Phase Flow”. In: *Annual Review of Fluid Mechanics* 15.1, pp. 261–291. ISSN: 1545-4479. DOI: 10.1146/annurev.fl.15.010183.001401.

Drew, D. A. and R. T. Lahey (1979). “Application of general constitutive principles to the derivation of multidimensional two-phase flow equations”. In: *International Journal of Multiphase Flow* 5.4, pp. 243–264. ISSN: 0301-9322. DOI: 10.1016/0301-9322(79)90024-7.

Drucker, D. and W. Prager (1952). “SOIL MECHANICS AND PLASTIC ANALYSIS OR LIMIT DESIGN”. In: *Quarterly of Applied Mathematics* 10.2, pp. 157–165. ISSN: 0033569X, 15524485.

Dumont, S. (2012). “On enhanced descent algorithms for solving frictional multicontact problems: application to the discrete element method”. In: *Int. J. Numer. Meth. Engng* 93.11, pp. 1170–1190. ISSN: 0029-5981. DOI: 10.1002/nme.4424.

Dunatunga, S. and K. Kamrin (2015). “Continuum modelling and simulation of granular flows through their many phases”. In: *Journal of Fluid Mechanics* 779, pp. 483–513. ISSN: 1469-7645. DOI: 10.1017/jfm.2015.383.

Duriez, C., F. Dubois, A. Kheddar, and C. Andriot (2006). “Realistic Haptic Rendering of Interacting Deformable Objects in Virtual Environments”. In: *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 1.12, pp. 36–47.

Duriez, C. (2008). “Rendering of Frictional Contact with Deformable Environments”. In: *Haptic Rendering: Foundations, Algorithms and Applications*. Natick, MA, USA: A. K. Peters, Ltd., pp. 421–442. ISBN: 1568813325.

Einstein, A. (1906). “Eine neue Bestimmung der Moleküldimensionen”. In: *Annalen der Physik* 324.2, pp. 289–306. ISSN: 1521-3889. DOI: 10.1002/andp.19063240204.

Ericson, C. (2004). *Real-time collision detection*. CRC Press.

Erleben, K. (2007). "Velocity-based shock propagation for multibody dynamics animation". In: *ACM Transaction on Graphics* 26.2.

Erleben, K. (2013). "Numerical methods for linear complementarity problems in physics-based animation". In: *ACM SIGGRAPH 2013 Courses on - SIGGRAPH'13*. DOI: 10.1145/2504435.2504443.

Etienne, J. (2004). "Simulation numérique d'écoulements gravitaires à fortes différences de densité : application aux avalanches". Thèse de doctorat dirigée par Saramito, Pierre et Hopfinger, Emil Mathématiques appliquées Grenoble, INPG 2004. PhD thesis, 163 p. URL: http://www.theses.fr/2004INPG0066.

Feng, Z.-Q., P. Joli, J.-M. Cros, and B. Magnain (2005). "The bi-potential method applied to the modeling of dynamic problems with friction". In: *Comput Mech* 36.5, pp. 375–383. ISSN: 1432-0924. DOI: 10.1007/s00466-005-0663-8.

Fischer, A. (1992). "A special newton-type optimization method". In: *Optimization* 24.3-4, pp. 269–284. ISSN: 1029-4945. DOI: 10.1080/02331939208843795.

Folgar, F. and C. L. Tucker (1984). "Orientation Behavior of Fibers in Concentrated Suspensions". In: *Journal of Reinforced Plastics and Composites* 3, pp. 98–119. DOI: 10.1177/073168448400300201.

Fortin, M. and R. Glowinski (1983). *Augmented Lagrangian Methods: Applications to the Numerical Solution of Boundary-Value Problems (Studies in Mathematics and Its Applications)*. Elsevier Science Ltd. ISBN: 0444866809.

Frankel, N. A. and A. Acrivos (1967). "On the viscosity of a concentrated suspension of solid spheres". In: *Chem. Eng. Sci.* 22, pp. 847–852.

Frigaard, I. and C. Nouar (2005). "On the usage of viscosity regularisation methods for visco-plastic fluid flow computation". In: *Journal of Non-Newtonian Fluid Mechanics* 127.1, pp. 1–26. ISSN: 0377-0257. DOI: 10.1016/j.jnnfm.2005.01.003.

Fukushima, M., Z.-Q. Luo, and P. Tseng (2002). "Smoothing Functions for Second-Order-Cone Complementarity Problems". In: *SIAM Journal on Optimization* 12 (2), pp. 436–460. ISSN: 1052-6234. DOI: http://dx.doi.org/10.1137/S1052623400380365.

GDR MiDi (2004). "On dense granular flows". In: *The European Physical Journal E* 14.4, pp. 341–365. ISSN: 1292-895X. DOI: 10.1140/epje/i2003-10153-0.

Glowinski, R. and A. Marroco (1975). "Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problèmes de Dirichlet non linéaires". fre. In: *ESAIM: Mathematical Modelling and Numerical Analysis* 9.R2, pp. 41–76. URL: http://eudml.org/doc/193269.

Goldstein, T., B. O'Donoghue, S. Setzer, and R. Baraniuk (2014). "Fast Alternating Direction Optimization Methods". In: *SIAM Journal on Imaging Sciences* 7.3, pp. 1588–1623. ISSN: 1936-4954. DOI: 10.1137/120896219.

Gornowicz, G. and S. Borac (2015). "Efficient and Stable Approach to Elasticity and Collisions for Hair Animation". In: *Proceedings of the 2015 Symposium on Digital Production*. DigiPro '15. Los Angeles, California: ACM, pp. 41–49. ISBN: 978-1-4503-3718-2. DOI: 10.1145/2791261.2791271.

Grinspun, E., A. N. Hirani, M. Desbrun, and P. Schröder (2003). "Discrete Shells". In: *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. SCA '03. San Diego, California: Eurographics Association, pp. 62–67. ISBN: 1-58113-659-5.

Grippo, L. and M. Sciandrone (2000). "On the convergence of the block nonlinear Gauss–Seidel method under convex constraints". In: *Operations Research Letters* 26.3, pp. 127–136. ISSN: 0167-6377. DOI: 10.1016/s0167-6377(99)00074-7.

Guennebaud, G., B. Jacob, et al. (2010). *Eigen v3, a C++ template library for linear algebra*. http://eigen.tuxfamily.org.

Hadap, S. (2006). "Oriented strands - dynamics of stiff multi-body system". In: *ACM SIGGRAPH - EG Symposium on Computer Animation (SCA'06)*. ACM-EG SCA, pp. 91–100.

Hadap, S. and N. Magnenat-Thalmann (2001). "Modeling Dynamic Hair as a Continuum". In: *Computer Graphics Forum* 20.3. Proceedings of Eurographics'01, pp. 329–338.

Haddouni, M. (2015). “Algorithmes de résolution de la dynamique du contact avec impact et frottement”. Thèse de doctorat dirigée par Brogliato, Bernard Matériaux, mécanique, génie civil, électrochimie Grenoble Alpes 2015. PhD thesis. URL: http://www.theses.fr/2015GREAI022.

Hanes, D. M. and A. J. Bowen (1985). “A granular-fluid model for steady intense bed-load transport”. In: *Journal of Geophysical Research* 90.C5, p. 9149. ISSN: 0148-0227. DOI: 10.1029/jc090ic05p09149.

Harlow, F. H. (1963). “The particle-in-cell method for numerical solution of problems in fluid dynamics”. In: *Proceedings of Symposia in Applied Mathematics*, pp. 269–288. ISSN: 0160-7634. DOI: 10.1090/psapm/015/9942. URL: http://dx.doi.org/10.1090/psapm/015/9942.

Harmon, D., E. Vouga, B. Smith, R. Tamstorf, and E. Grinspun (2009). “Asynchronous Contact Mechanics”. In: *ACM Trans. Graph.* 28.3, 87:1–87:12. ISSN: 0730-0301. DOI: 10.1145/1531326.1531393.

Harmon, D., E. Vouga, R. Tamstorf, and E. Grinspun (2008). “Robust treatment of simultaneous collisions”. In: *ACM Transactions on Graphics* 27.3, p. 1. ISSN: 0730-0301. DOI: 10.1145/1360612.1360622.

Harris, S. E. and D. G. Crighton (1994). “Solitons, solitary waves, and voidage disturbances in gas-fluidized beds”. In: *Journal of Fluid Mechanics* 266, pp. 243–276. ISSN: 1469-7645. DOI: 10.1017/S0022112094000996.

Haslinger, J. (1983). “Approximation of the signorini problem with friction, obeying the coulomb law”. In: *Mathematical Methods in the Applied Sciences* 5.1, pp. 422–437. ISSN: 0170-4214. DOI: 10.1002/mma.1670050127.

Herrera, T. L., A. Zinke, and A. Weber (2012). “Lighting Hair From The Inside: A Thermal Approach To Hair Reconstruction”. In: *ACM Transactions on Graphics (Proceedings SIGGRAPH Asia 2012)*. Vol. 31. 6. DOI: 10.1145/2366145.2366165.

Heyman, J., R. Delannay, H. Tabuteau, and A. Valance (2016). “Compressibility regularizes the "$\mu(I)$" rheology for granular flows”. In: eprint: 1609.01502.

Heyn, T. D. (2013). “On the modeling, simulation, and visualization of many-body dynamics problems with friction and contact”. PhD thesis. THE UNIVERSITY OF WISCONSIN-MADISON.

Hiriart-Urruty, J.-B. and C. Lemaréchal (1993). *Convex analysis and minimization algorithms I: Fundamentals*. Vol. 305. Springer Science & Business Media.

Hsu, T.-J., J. T. Jenkins, and P. L.-. F. Liu (2004). “On two-phase sediment transport: sheet flow of massive particles”. In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 460.2048, pp. 2223–2250. ISSN: 1471-2946. DOI: 10.1098/rspa.2003.1273.

Iben, H., M. Meyer, L. Petrovic, O. Soares, et al. (2013). “Artistic Simulation of Curly Hair”. In: *Proceedings of the 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. SCA ’13. Anaheim, California: ACM, pp. 63–71. ISBN: 978-1-4503-2132-7. DOI: 10.1145/2485895.2485913.

Ihmsen, M., A. Wahl, and M. Teschner (2013). “A Lagrangian framework for simulating granular material with high detail”. In: *Computers & Graphics* 37.7, pp. 800–808. ISSN: 0097-8493. DOI: 10.1016/j.cag.2013.04.010.

Ionescu, I. R. (2010). “Onset and dynamic shallow flow of a viscoplastic fluid on a plane slope”. In: *Journal of Non-Newtonian Fluid Mechanics* 165.19-20, pp. 1328–1341. ISSN: 0377-0257. DOI: 10.1016/j.jnnfm.2010.06.016.

— (2013). “Augmented Lagrangian for shallow viscoplastic flow with topography”. In: *Journal of Computational Physics* 242, pp. 544–560. ISSN: 0021-9991. DOI: 10.1016/j.jcp.2013.02.029.

Ionescu, I. R., A. Mangeney, F. Bouchut, and O. Roche (2015). “Viscoplastic modeling of granular column collapse with pressure-dependent rheology”. In: *Journal of Non-Newtonian Fluid Mechanics* 219, pp. 1–18. ISSN: 0377-0257. DOI: 10.1016/j.jnnfm.2015.02.006.

Iverson, R. M. (1997). “The physics of debris flows”. In: *Reviews of Geophysics* 35.3, pp. 245–296. ISSN: 8755-1209. DOI: 10.1029/97rg00426.

Jackson, R. (1963). "The mechanics of fluidised beds. Part I. The stability of the state of uniform fluidisation". In: *Transactions of the Institution of Chemical Engineers*.

Jackson, R. (2000). *The dynamics of fluidized particles*. Cambridge University Press.

Jean, M. (1999). "The non-smooth contact dynamics method". In: *Computer Methods in Applied Mechanics and Engineering* 177.3-4, pp. 235–257. ISSN: 0045-7825. DOI: 10.1016/s0045-7825(98)00383-1.

Jean, M. and J.-J. Moreau (1987). "Dynamics in the Presence of Unilateral Contacts and Dry Friction: A Numerical Approach". In: *Unilateral Problems in Structural Analysis — 2: Proceedings of the Second Meeting on Unilateral Problems in Structural Analysis, Prescudin, June 17–20, 1985*. Ed. by G. Del Piero and F. Maceri. Vienna: Springer Vienna, pp. 151–196. ISBN: 978-3-7091-2967-8. DOI: 10.1007/978-3-7091-2967-8_10.

— (1992). "Unilaterality and dry friction in the dynamics of rigid bodies collections". In: *Proceedings of the Contact Mechanics International Symposium*. Ed. by A. Curnier, pp. 31–48.

Jeffery, G. B. (1922). "The Motion of Ellipsoidal Particles Immersed in a Viscous Fluid". In: *Proceedings of the Royal Society of London. Series A* 102, pp. 161–179. DOI: 10.1098/rspa.1922.0078.

Jiang, C., C. Schroeder, A. Selle, J. Teran, and A. Stomakhin (2015). "The Affine Particle-in-cell Method". In: *ACM Transactions on Graphics* 34.4, pp. 1–10. ISSN: 0730-0301. DOI: 10.1145/2766996.

Jiang, C., C. Schroeder, J. Teran, A. Stomakhin, and A. Selle (2016). "The material point method for simulating continuum materials". In: *ACM SIGGRAPH 2016 Courses on - SIGGRAPH '16*. DOI: 10.1145/2897826.2927348.

Jop, P., Y. Forterre, and O. Pouliquen (2006). "A constitutive law for dense granular flows". In: *Nature* 441.7094, pp. 727–730. ISSN: 1476-4687. DOI: 10.1038/nature04801.

Jourdan, F., P. Alart, and M. Jean (1998). "A Gauss-Seidel like algorithm to solve frictional contact problems". In: *Computer Methods in Applied Mechanics and Engineering* 155, pp. 31–47. DOI: 10.1016/S0045-7825(97)00137-0.

Kaufman, D. M., S. Sueda, D. James, and D. Pai (2008). "Staggered Projections for Frictional Contact in Multibody Systems". In: *ACM Transactions on Graphics (Proceedings of the ACM SIGGRAPH Asia'08 conference)* 27.5, 164:1–164:11.

Kaufman, D. M., R. Tamstorf, B. Smith, J.-M. Aubry, and E. Grinspun (2014). "Adaptive nonlinearity for collisions in complex rod assemblies". In: *ACM Transactions on Graphics* 33.4, pp. 1–12. ISSN: 0730-0301. DOI: 10.1145/2601097.2601100.

Klar, G., T. Gast, A. Pradhana, C. Fu, et al. (2016). "Drucker-Prager Elastoplasticity for Sand Animation". In: *ACM Trans. Graph.* 35.4.

Klarbring, A. (1987). "Contact Problems with Friction by Linear Complementarity". In: *Unilateral Problems in Structural Analysis – 2*, pp. 197–219. DOI: 10.1007/978-3-7091-2967-8_11.

— (1990). "Examples of non-uniqueness and non-existence of solutions to quasistatic contact problems with friction". In: *Ingenieur-Archiv* 60.8, pp. 529–541. ISSN: 1432-0681. DOI: 10.1007/BF00541909.

Konagai, K. and J. Johansson (2001). "Two dimensional Lagrangian Particle Finite Difference Method for modeling large soil deformations". In: *Journal of Structural Mechanics and Earthquake Engineering* 18, pp. 105–110.

Krieger, I. M. (1972). "Rheology of monodisperse latices". In: *Advances in Colloid and Interface Science* 3.2, pp. 111–136. ISSN: 0001-8686. DOI: 10.1016/0001-8686(72)80001-0.

Kynch, G. J. (1952). "A theory of sedimentation". In: *Transactions of the Faraday society* 48, pp. 166–176.

Lagrée, P.-Y., L. Staron, and S. Popinet (2011). "The granular column collapse as a continuum: validity of a two-dimensional Navier–Stokes model with a $\mu(I)$-rheology". In: *Journal of Fluid Mechanics* 686, pp. 378–408. ISSN: 1469-7645. DOI: 10.1017/jfm.2011.335.

Lajeunesse, E., J. B. Monnier, and G. M. Homsy (2005). "Granular slumping on a horizontal surface". In: *Physics of Fluids* 17.10, p. 103302. ISSN: 1070-6631. DOI: 10.1063/1.2087687.

Lenaerts, T., B. Adams, and P. Dutré (2008). "Porous flow in particle-based fluid simulations". In: *ACM SIGGRAPH 2008 papers on - SIGGRAPH '08*. DOI: 10.1145/1399504.1360648.

Lenaerts, T. and P. Dutré (2009). “Mixing fluids and granular materials”. In: *Computer Graphics Forum*. Vol. 28. 2, pp. 213–218. DOI: 10.1111/j.1467-8659.2009.01360.x.

Mankoc, C., A. Janda, R. Arévalo, J. M. Pastor, et al. (2007). “The flow rate of granular materials through an orifice”. In: *Granular Matter* 9.6, pp. 407–414. ISSN: 1434-7636. DOI: 10.1007/s10035-007-0062-2.

Manninen, M., V. Taivassalo, and S. Kallio (1996). “On the Mixture model for Multiphase Flow”. In: *VTT Publications* 288. URL: http://www.vtt.fi/inf/pdf/publications/1996/P288.pdf.

Mast, C. M. (2013). “Modeling landslide-induced flow interactions with structures using the material point method”. PhD thesis. University of Washington.

Mast, C. M., P. Arduino, P. Mackenzie-Helnwein, and G. R. Miller (2014). “Simulating granular column collapse using the Material Point Method”. In: *Acta Geotechnica* 10.1, pp. 101–116. ISSN: 1861-1133. DOI: 10.1007/s11440-014-0309-0.

Maurin, R., J. Chauchat, B. Chareyre, and P. Frey (2015). “A minimal coupled fluid-discrete element model for bedload transport”. In: *Phys. Fluids* 27.11, p. 113302. ISSN: 1089-7666. DOI: 10.1063/1.4935703.

Mazhar, H., T. Heyn, D. Negrut, and A. Tasora (2015). “Using Nesterov’s Method to Accelerate Multibody Dynamics with Friction and Contact”. In: *ACM Transactions on Graphics* 34.3, pp. 1–14. ISSN: 0730-0301. DOI: 10.1145/2735627.

McAdams, A., A. Selle, K. Ward, E. Sifakis, and J. Teran (2009). “Detail preserving continuum simulation of straight hair”. In: *ACM Transactions on Graphics Proceedings of the SIGGRAPH’09 conference)*. Association for Computing Machinery, p. 1. ISBN: 9781605587264. DOI: 10.1145/1576246.1531368.

Michaels, A. S. and J. C. Bolger (1962). “Settling Rates and Sediment Volumes of Flocculated Kaolin Suspensions”. In: *Industrial & Engineering Chemistry Fundamentals* 1.1, pp. 24–33. ISSN: 1541-4833. DOI: 10.1021/i160001a004.

Miguel, E., R. Tamstorf, D. Bradley, S. C. Schvartzman, et al. (2013). “Modeling and Estimation of Internal Friction in Cloth”. In: *ACM Trans. Graph.* 32.6, 212:1–212:10. ISSN: 0730-0301. DOI: 10.1145/2508363.2508389.

Moreau, J.-J. (1965). “Proximité et dualité dans un espace hilbertien”. fre. In: *Bulletin de la Société Mathématique de France* 93, pp. 273–299. URL: http://eudml.org/doc/87067.

— (1966–1967). “Fonctionnelles convexes”. fre. In: *Séminaire Jean Leray* 2, pp. 1–108. URL: http://eudml.org/doc/112529.

— (1988). “Unilateral Contact and Dry Friction in Finite Freedom Dynamics”. In: *Nonsmooth Mechanics and Applications*. Ed. by J.-J. Moreau and P. D. Panagiotopoulos. Vienna: Springer Vienna, pp. 1–82. ISBN: 978-3-7091-2624-0. DOI: 10.1007/978-3-7091-2624-0_1.

— (1999). “Numerical aspects of the sweeping process”. In: *Computer Methods in Applied Mechanics and Engineering* 177.3–4, pp. 329–349. ISSN: 0045-7825. DOI: http://dx.doi.org/10.1016/S0045-7825(98)00387-9.

Müller, M., B. Solenthaler, R. Keiser, and M. Gross (2005). “Particle-based fluid-fluid interaction”. In: *Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation - SCA ’05*. DOI: 10.1145/1073368.1073402.

Munson, T. S., F. Facchinei, M. C. Ferris, A. Fischer, and C. Kanzow (2001). “The semismooth algorithm for large scale complementarity problems”. In: *INFORMS Journal on Computing* 13.4, pp. 294–311.

Narain, R., A. Golas, and M. C. Lin (2010). “Free-flowing granular materials with two-way solid coupling”. In: *ACM Transactions on Graphics* 29.6, pp. 1–10. DOI: 10.1145/1882261.1866195.

Narain, R., M. Overby, and G. E. Brown (2016). “ADMM ⊇ Projective Dynamics: Fast Simulation of General Constitutive Models”. In: *Eurographics/ ACM SIGGRAPH Symposium on Computer Animation*. Ed. by L. Kavan and C. Wojtan. The Eurographics Association. ISBN: 978-3-03868-009-3. DOI: 10.2312/sca.20161219.

Nesterov, Y. (1983). “A method of solving a convex programming problem with convergence rate $O(1/k^2)$”. In: *Soviet Mathematics Doklady* 27.2, pp. 372–376.

Ngo Ngoc, C. and S. Boivin (2004). *Nonlinear Cloth Simulation*. Tech. rep. Inria.

Nielsen, M. B. and O. Østerby (2013). "A two-continua approach to Eulerian simulation of water spray". In: *ACM Transactions on Graphics* 32.4, p. 1. ISSN: 0730-0301. DOI: 10.1145/2461912.2461918.

Otaduy, M. A., R. Tamstorf, D. Steinemann, and M. Gross (2009). "Implicit Contact Handling for Deformable Objects". In: *Computer Graphics Forum* 28.2, pp. 559–568. ISSN: 1467-8659. DOI: 10.1111/j.1467-8659.2009.01396.x.

Pai, D. (2002). "Strands: Interactive Simulation of Thin Solids using Cosserat Models". In: *Computer Graphics Forum* 21.3. Proceedings of Eurographics'02, pp. 347–352.

Pailha, M., M. Nicolas, and O. Pouliquen (2008). "Initiation of underwater granular avalanches: Influence of the initial volume fraction". In: *Physics of Fluids* 20.11, p. 111701. ISSN: 1070-6631. DOI: 10.1063/1.3013896.

Parikh, N. and S. P. Boyd (2014). "Proximal Algorithms." In: *Foundations and Trends in optimization* 1.3, pp. 127–239.

Parker, G., Y. Fukushima, and H. M. Pantin (1986). "Self-accelerating turbidity currents". In: *Journal of Fluid Mechanics* 171.-1, p. 145. ISSN: 1469-7645. DOI: 10.1017/s0022112086001404.

Pellegrini, F. and J. Roman (1996). "Scotch: A software package for static mapping by dual recursive bipartitioning of process and architecture graphs". In: *Lecture Notes in Computer Science*, pp. 493–498. ISSN: 1611-3349. DOI: 10.1007/3-540-61142-8_588.

Pietro, D. A. D. and A. Ern (2011). *Mathematical Aspects of Discontinuous Galerkin Methods (Mathématiques et Applications)*. Springer. ISBN: 3642229794.

Pignatel, F., M. Nicolas, É. Guazzelli, and D. Saintillan (2009). "Falling jets of particles in viscous fluids". In: *Physics of Fluids* 21.12, p. 123303. ISSN: 1070-6631. DOI: 10.1063/1.3276235.

Pitman, E. B. and L. Le (2005). "A two-fluid model for avalanche and debris flows". In: *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 363.1832, pp. 1573–1601. ISSN: 1471-2962. DOI: 10.1098/rsta.2005.1596.

Plante, E., M.-P. Cani, and P. Poulin (2001). "A Layered Wisp Model for Simulating Interactions inside Long Hair". In: *EG workshop on Computer Animation and Simulation (EG CAS'01)*. Computer Science. Springer, pp. 139–148.

Potra, F. A. and S. J. Wright (2000). "Interior-point methods". In: *Journal of Computational and Applied Mathematics* 124.1-2, pp. 281–302. ISSN: 0377-0427. DOI: 10.1016/s0377-0427(00)00433-7.

Qi, L. and J. Sun (1993). "A nonsmooth version of Newton's method". In: *Mathematical Programming* 58, pp. 353–367.

Radjai, F., D. Wolf, M. Jean, and J.-J. Moreau (1998). "Bimodal Character of Stress Transmission in Granular Packings". In: *Phys. Rev. Lett.* 80 (1), pp. 61–64. DOI: 10.1103/PhysRevLett.80.61.

Ram, D., T. Gast, C. Jiang, C. Schroeder, et al. (2015). "A material point method for viscoelastic fluids, foams and sponges". In: *Proceedings of the 14th ACM SIGGRAPH / Eurographics Symposium on Computer Animation - SCA '15*. DOI: 10.1145/2786784.2786798.

Raous, M., P. Chabrand, and F. Lebon (1988). "Numerical methods for frictional contact problems and applications". In: *Journal de Mécanique Théorique et Appliquée* 7.1.

Ren, B., C. Li, X. Yan, M. C. Lin, et al. (2014). "Multiple-Fluid SPH Simulation Using a Mixture Model". In: *ACM Transactions on Graphics* 33.5, pp. 1–11. ISSN: 0730-0301. DOI: 10.1145/2645703.

Renouf, M., V. Acary, and G. Dumont (2005). "3D Frictional Contact and Impact Multibody Dynamics. A Comparison of Algorithms Suitable for Real-time Applications". In: *ECCOMAS*.

Renouf, M. and P. Alart (2005). "Conjugate gradient type algorithms for frictional multi-contact problems: applications to granular materials". In: *Computer Methods in Applied Mechanics and Engineering* 194.18-20, pp. 2019–2041. ISSN: 0045-7825. DOI: 10.1016/j.cma.2004.07.009.

Revil-Baudard, T. and J. Chauchat (2013). "A two-phase model for sheet flow regime based on dense granular flow rheology". In: *Journal of Geophysical Research: Oceans* 118.2, pp. 619–634. ISSN: 2169-9275. DOI: 10.1029/2012jc008306.

Richardson, J. F. and W. N. Zaki (1954). “Sedimentation and fluidisation: Part I. Transactions of Institution of Chemical Engineers”. In: *Transactions of Institution of Chemical Engineers* 32, pp. 35–53.

Rockafellar, R. T. (1993). “Lagrange Multipliers and Optimality”. In: *SIAM Review* 35.2, pp. 183–238. ISSN: 00361445. DOI: 10.2307/2133143.

Rosenblum, R., W. Carlson, and E. Tripp (1991). “Simulating the Structure and Dynamics of Human Hair: Modeling, Rendering, and Animation”. In: *The Journal of Visualization and Computer Animation* 2.4, pp. 141–148.

Roux, S. and F. Radjai (1998). “Texture-Dependent Rigid-Plastic Behavior”. In: *Physics of Dry Granular Media*, pp. 229–236. DOI: 10.1007/978-94-017-2653-5_13.

Rungjiratananon, W., Z. Szego, Y. Kanamori, and T. Nishita (2008). “Real-time Animation of Sand-Water Interaction”. In: *Computer Graphics Forum* 27.7, pp. 1887–1893. ISSN: 1467-8659. DOI: 10.1111/j.1467-8659.2008.01336.x.

Saad, Y. (2003). *Iterative Methods for Sparse Linear Systems, Second Edition*. Society for Industrial and Applied Mathematics. ISBN: 0898715342.

Saramito, P. (2013). “Méthodes numériques en fluides complexes : théorie et algorithmes”. Lecture. Grenoble. URL: https://cel.archives-ouvertes.fr/cel-00673816.

— (2015). “Efficient C++ finite element computing with Rheolef”. Lecture. Grenoble, France, France. URL: https://cel.archives-ouvertes.fr/cel-00573970.

— (2016). “A damped Newton algorithm for computing viscoplastic fluid flows”. In: *Journal of Non-Newtonian Fluid Mechanics*. ISSN: 0377-0257. DOI: 10.1016/j.jnnfm.2016.05.007.

Saramito, P. and N. Roquet (2001). “An adaptive finite element method for viscoplastic fluid flows in pipes”. In: *Computer Methods in Applied Mechanics and Engineering* 190.40-41, pp. 5391–5412. ISSN: 0045-7825. DOI: 10.1016/s0045-7825(01)00175-x.

Saramito, P. and A. Wachs (2016). “Progress in numerical simulation of yield stress fluid flows”. working paper or preprint. URL: https://hal.archives-ouvertes.fr/hal-01375720.

Savage, S. B. and K. Hutter (1989). “The motion of a finite mass of granular material down a rough incline”. In: *Journal of Fluid Mechanics* 199.-1, p. 177. ISSN: 1469-7645. DOI: 10.1017/s0022112089000340.

Seguin, A., C. Coulais, F. Martinez, Y. Bertho, and P. Gondret (2016). “Local rheological measurements in the granular flow around an intruder”. In: *Physical Review E* 93.1. ISSN: 2470-0053. DOI: 10.1103/physreve.93.012904.

Selle, A., M. Lentine, and R. Fedkiw (2008). “A mass spring model for hair simulation”. In: *ACM Transactions on Graphics (Proceedings of the ACM SIGGRAPH'08 conference)* 27.3, pp. 1–11. ISSN: 0730-0301. DOI: http://doi.acm.org/10.1145/1360612.1360663.

Shu, C.-W. (2009). “High Order Weighted Essentially Nonoscillatory Schemes for Convection Dominated Problems”. In: *SIAM Review* 51.1, pp. 82–126. ISSN: 1095-7200. DOI: 10.1137/070679065.

Sigal, L., M. Mahler, S. Diaz, K. McIntosh, et al. (2015). “A perceptual control space for garment simulation”. In: *ACM Transactions on Graphics* 34.4, 117:1–117:10. ISSN: 0730-0301. DOI: 10.1145/2766971.

Silcowitz, M., S. Niebe, and K. Erleben (2009). “Nonsmooth Newton Method for Fischer function reformulation of contact force problems for interactive rigid body simulation”. In: pp. 105–114. DOI: 10.2312/PE/vriphys/vriphys09/105-114.

Simo, J. and T. Hughes (2000). *Computational Inelasticity (Interdisciplinary Applied Mathematics) (v. 7)*. Springer. ISBN: 0387975209.

Smith, B., D. M. Kaufman, E. Vouga, R. Tamstorf, and E. Grinspun (2012). “Reflections on Simultaneous Impact”. In: *ACM Trans. Graph.* 31.4, 106:1–106:12. ISSN: 0730-0301. DOI: 10.1145/2185520.2185602.

Spillmann, J. and M. Teschner (2007). “CoRdE: Cosserat rod elements for the dynamic simulation of one-dimensional elastic objects”. In: *ACM SIGGRAPH - EG Symposium on Computer Animation (SCA'07)*. ACM-EG SCA, pp. 63–72.

Staron, L. and E. J. Hinch (2005). "Study of the collapse of granular columns using two-dimensional discrete-grain simulation". In: *Journal of Fluid Mechanics* 545.-1, p. 1. ISSN: 1469-7645. DOI: 10.1017/s0022112005006415.

Staron, L., P.-Y. Lagrée, and S. Popinet (2012). "The granular silo as a continuum plastic flow: The hour-glass vs the clepsydra". In: *Physics of Fluids* 24, 103301. ISSN: 1070-6631. DOI: 10.1063/1.4757390.

— (2014). "Continuum simulation of the discharge of the granular silo". In: *The European Physical Journal E* 37.1. ISSN: 1292-895X. DOI: 10.1140/epje/i2014-14005-6.

Steffen, M., R. M. Kirby, and M. Berzins (2008). "Analysis and reduction of quadrature errors in the material point method (MPM)". In: *Int. J. Numer. Meth. Engng* 76.6, pp. 922–948. ISSN: 1097-0207. DOI: 10.1002/nme.2360.

Stomakhin, A., C. Schroeder, L. Chai, J. Teran, and A. Selle (2013). "A material point method for snow simulation". In: *ACM Transactions on Graphics* 32.4, pp. 1–10. DOI: 10.1145/2461912.2461948.

Stomakhin, A., C. Schroeder, C. Jiang, L. Chai, et al. (2014). "Augmented MPM for phase-change and varied materials". In: *ACM Transactions on Graphics* 33.4, pp. 1–11. ISSN: 0730-0301. DOI: 10.1145/2601097.2601176.

Sulsky, D. (1994). "A particle method for history-dependent materials". In: *Computer Methods in Applied Mechanics and Engineering* 118, pp. 179–196. DOI: 10.1016/0045-7825(94)00033-6.

Sun, D. and J. Sun (2005). "Strong Semismoothness of the Fischer-Burmeister SDC and SOC Complementarity Functions". In: *Math. Program.* 103.3, pp. 575–581. ISSN: 1436-4646. DOI: 10.1007/s10107-005-0577-4.

Tasora, A. (2013). "Efficient simulation of contacts, friction and constraints using a modified spectral projected gradient method". In: *Poster Proceedings of WSCG 2013*, pp. 69–72.

Terzaghi, K. (1936). "The Shearing Resistance of Saturated Soils". In: *Proc. 1st International Conference on Soil Mechanics and Foundation Engineering*.

Teschner, M., B. Heidelberger, M. Müller, D. Pomeranerts, and M. Gross (2003). "Optimized Spatial Hashing for Collision Detection of Deformable Objects". In: *Vision, Modeling, Visualization (VMV 2003)*, pp. 47–54.

Tonge, R., F. Benevolenski, and A. Voroshilov (2012). "Mass splitting for jitter-free parallel rigid body simulation". In: *ACM Transactions on Graphics* 31.4, pp. 1–8. ISSN: 0730-0301. DOI: 10.1145/2185520.2185601.

Topin, V., F. Dubois, Y. Monerie, F. Perales, and A. Wachs (2011). "Micro-rheology of dense particulate flows: Application to immersed avalanches". In: *Journal of Non-Newtonian Fluid Mechanics* 166.1-2, pp. 63–72. ISSN: 0377-0257. DOI: 10.1016/j.jnnfm.2010.10.006.

Topin, V., Y. Monerie, F. Perales, and F. Radjaï (2012). "Collapse Dynamics and Runout of Dense Granular Materials in a Fluid". In: *Physical Review Letters* 109.18. ISSN: 1079-7114. DOI: 10.1103/physrevlett.109.188001.

Treskatis, T., M. A. Moyers-González, and C. J. Price (2016). "An accelerated dual proximal gradient method for applications in viscoplasticity". In: *Journal of Non-Newtonian Fluid Mechanics*. ISSN: 0377-0257. DOI: 10.1016/j.jnnfm.2016.09.004.

Trinkle, J. C., J.-S. Pang, S. Sudarsky, and G. Lo (1997). "On Dynamic Multi-Rigid-Body Contact Problems with Coulomb Friction". In: *Z. angew. Math. Mech.* 77.4, pp. 267–279. ISSN: 1521-4001. DOI: 10.1002/zamm.19970770411.

Ungarish, M. (2009). *An introduction to gravity currents and intrusions*. CRC Press.

Vermeer, P. A. (1998). "Non-Associated Plasticity for Soils, Concrete and Rock". In: *Physics of Dry Granular Media*, pp. 163–196. DOI: 10.1007/978-94-017-2653-5_10.

Ward, K., F. Bertails, T.-Y. Kim, S. Marschner, et al. (2007). "A Survey on Hair Modeling: Styling, Simulation, and Rendering". In: *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 13.2, pp. 213–34. URL: http://www-evasion.imag.fr/Publications/2007/WBKMCL07.

Wieckowski, Z. and M. Pawlak (2015). "Material point method in three-dimensional problems of granular flow". In: *6th European Conference on Computational Fluid Dynamics*.

Wieckowski, Z., S.-K. Youn, and J.-H. Yeon (1999). “A particle-in-cell solution to the silo discharging problem”. In: *International Journal for Numerical Methods in Engineering* 45.9, pp. 1203–1225. ISSN: 1097-0207. DOI: 10.1002/(sici)1097-0207(19990730)45:9<1203::aid-nme626>3.0.co;2-c.

Yue, Y., B. Smith, C. Batty, C. Zheng, and E. Grinspun (2015). “Continuum Foam”. In: *ACM Transactions on Graphics* 34.5, pp. 1–20. ISSN: 0730-0301. DOI: 10.1145/2751541.

Zhu, H., X. Liu, Y. Liu, and E. Wu (2006). “Simulation of miscible binary mixtures based on lattice Boltzmann method”. In: *Computer Animation and Virtual Worlds* 17.3-4, pp. 403–410. ISSN: 1546-427X. DOI: 10.1002/cav.143.

Zhu, Y. and R. Bridson (2005). “Animating sand as a fluid”. In: *ACM Transactions on Graphics* 24.3, pp. 965–972. DOI: 10.1145/1073204.1073298.

Abstract

This dissertation focuses on the numerical simulation of mechanical systems that consist of a large number of discrete pieces interacting with each other through contacts and dry friction. Examples of such systems — for instance, sand or human hair — are common in natural environments; being able to predict their dynamics is therefore of importance for diverse applications, ranging from geotechnical considerations to engaging visual effects for feature films.

A major difficulty that complicates the numerical simulation of such complex systems stems from the nonsmoothness of their dynamics. Indeed, from a macroscopic viewpoint, the velocity of the individual constituents may exhibit instantaneous jumps, for instance when impacts occur. The first part of this manuscript will thus be dedicated to the establishment of efficient algorithms for the numerical simulation of discrete mechanical systems subject to contacts and Coulomb friction. We advocate using a Gauss–Seidel algorithm with an hybrid local solver, and show that this strategy performs robustly in the challenging context of virtual hair simulation.

The second part of this dissertation will focus on much bigger systems, consisting of millions or billions of grains. As computing every force between pairs of contacting grains quickly becomes intractable, we embrace a continuum viewpoint instead. We show how the numerical methods devised for the simulation of discrete mechanical systems can be adapted to the simulation of flows governed by the Drucker–Prager rheology — a constitutive relationship between the material's stress and strain rate that macroscopically models the action of frictional contact forces. This approaches allows us to capture experimentally-observed qualitative features of granular flows. Finally, we propose a new numerical model for the coupled simulation of a granular continuum with a surrounding fluid, and show that once again, we are able to leverage algorithms from discrete contact mechanics.

Résumé

Cette thèse traite de la simulation numérique de systèmes composés de nombreux objets distincts, et dont le principal mécanisme d'interaction consiste en des contacts inélastiques avec frottement sec. On trouve de nombreuses occurrences de tels systèmes dans la nature, par exemple sous la forme de sable ou d'une chevelure humaine ; aussi la reproduction numérique de leur dynamique trouve des applications diverses, allant de considérations géotechniques à la production d'effets spéciaux réalistes.

Une difficulté majeure pour la simulation de tels systèmes concerne la *non-régularité* de leur dynamique ; à une échelle de temps macroscopique, on observe des sauts dans les vitesses des constituants, par exemple lors d'impacts. La première partie de ce manuscrit est ainsi dédiée à la conception d'algorithmes efficaces permettant de prendre en compte les contacts avec frottement de Coulomb lors de la simulation de systèmes mécaniques discrets. La méthode que nous proposons, basée sur un algorithme de type Gauss–Seidel avec stratégie hybride, s'avère robuste et perfomante sur le problème délicat de la simulation virtuelle de chevelures.

La second partie de ce manuscrit est consacrée à l'étude de systèmes à une échelle beaucoup plus grande, au delà du million de grains. Puisque le calcul de toutes les forces de contacts pour chaque paire de grains s'avérerait trop coûteux, on adopte un point de vue macroscopique. On propose ainsi d'adapter les méthodes développées pour la simulation de systèmes discrets à la résolution de la rhéologie dite de Drucker–Prager, une relation entre la contrainte et le cisaillement du matériau exprimant l'influence moyennée des forces de frottement. On montre que cette approche nous permet de retrouver le comportement qualitatif de matériaux granulaires secs observé expérimentalement. Finalement, nous proposons un nouveau modèle numérique pour l'étude des dynamiques couplées d'un matériau granulaire immergé dans un fluide Newtonien, et montrons une nouvelle fois que les algorithmes développés pour la mécanique discrète s'avèrent également pertinents dans le cas continu.