Abaques virtuelle pour le génie parasismique incluant des parametres associes au chargement

Sebastian Rodriguez Iturra

Thèse de doctorat

NNT: 2021UPAST097

université PARIS-SACLAY

Abaques virtuels pour le génie parasismique incluant des paramètres associés au chargement

Virtual charts for earthquake engineering including loading parameters

Thèse de doctorat de l'Université Paris-Saclay

École doctorale n° 579, Sciences Mécaniques et Energétiques, Matériaux et Géosciences (SMEMAG)
Spécialité de doctorat: Génie civil
Unité de recherche: Université Paris-Saclay, ENS Paris-Saclay, CNRS, LMT - Laboratoire de Mécanique et Technologie, 91190, Gif-sur-Yvette, France
Référent: ENS Paris-Saclay

Thèse présentée et soutenue à Paris-Saclay, le 18 Octobre 2021, par

Sebastian Rodriguez Iturra

Composition du jury:

David Dureisseix Professeur, INSA Lyon	Président
Delphine Brancherie Professeur, Université de Technologie de Compiègne	Rapportrice
Elias Cueto Professeur, Université de Saragosse	Rapporteur
Andrea Barbarulo Maître de conférences, CentraleSupélec	Examinateur
Giuseppe Abbiati Maître de conférences, Université d'Aarhus	Examinateur
David Néron Professeur, École normale supérieure Paris-Saclay	Directeur
Pierre Ladevèze Professeur émérite, École normale supérieure Paris-Saclay	Co-directeur
Pierre-Etienne Charbonnel Ingénieur chercheur, CEA	Co-encadrant
Georges Nahas Ingénieur chercheur, IRSN	Invité

université PARIS-SACLAY
ÉCOLE DOCTORALE
Sciences mécaniques et énergétiques, matériaux et géosciences (SMEMAG)

Titre: Abaques virtuels pour le génie parasismique incluant des paramètres associés au chargement

Mots clés: Méthode à grand incrément de temps pour la dynamique non linéaire à basse fréquence, Décomposition Propre Généralisée, Méthode Galerkin discontinue en temps, Théorie du signal - méthode multi-échelle temporelle, Méthode d'hyper-réduction, Stratégie parallèle en temps.

Résumé: La complexité et la finesse des modèles numériques utilisés pour prédire le comportement sismique (souvent non linéaire) des structures en béton armé imposent un temps de calcul de plusieurs jours pour résoudre les équations aux dérivées partielles du problème de référence. De plus, pour l'évaluation des marges, les analyses de sécurité ou la mise à jour des modèles, la prise en compte des incertitudes associées aux paramètres constitutifs ou au chargement sismique lui-même impose généralement de prévoir cet effort numérique, non pas pour la simulation d'un seul modèle, mais d'une famille de modèles soumis à un ensemble d'entrées probables définissant le scénario sismique. Pour réduire les temps de calcul, certaines techniques, appelées "réduction de l'ordre des modèles", doivent être envisagées. La méthode à grand incrément de temps (LATIN dans son acronyme anglais) utilisée en combinaison avec la technique de réduction de l'ordre des modèles appelée Décomposition Propre Généralisée (PGD dans son acronyme anglais) a prouvé son efficacité pour la résolution de problèmes non linéaires en mécanique. Jusqu'à présent, la méthode LATIN-PGD n'a jamais été appliquée à la résolution de problèmes non linéaires en dynamique. Dans ce contexte, le cadre de LATIN-PGD est d'abord adapté au cas dynamique, où les non-linéarités considérées correspondent aux matériaux typiques du béton armé, c'est-à-dire l'endommagement isotrope quasi-fragile pour le béton et l'élasto-viscoplasticité pour les métaux; de plus, des stratégies dédiées sont développées pour réduire les coûts de calcul lors de la prise en compte d'excitations complexes et de grande durée, telles que les sollicitations sismiques ou de fatigue. Les contributions de ce travail de thèse sont les suivantes: (i) une adaptation de la méthode de Galerkin discontinue dans le temps (TDGM dans son acronyme anglais) pour résoudre les problèmes temporels incrémentaux dans le cadre de LATIN-PGD, qui permet de résoudre efficacement les problèmes où l'intervalle de temps est relativement grand, (ii) une nouvelle technique d'approximation du signal et une nouvelle stratégie multi-échelle dans le temps sont développées pour optimiser la résolution des fonctions PGD temporelles lors du traitement d'excitations à long terme telles que les entrées sismiques ou les charges de fatigue; (iii) une technique d'hyper-réduction est proposée pour accélérer la construction de l'approximation PGD à faible rang et enfin, On introduit une stratégie de résolution en temps parallèle basée sur l'utilisation de TDGM, qui vise à accélérer la résolution temporelle de la PGD en tirant parti des architectures parallèles des ordinateurs récents. Toutes les contributions précédentes permettent une forte optimisation du cadre LATIN-PGD, ce qui permet par conséquent une réduction considérable du coût numérique pour obtenir la réponse non linéaire d'une structure en dynamique.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France

université PARIS-SACLAY

ÉCOLE DOCTORALE
Sciences mécaniques et énergétiques, matériaux et géosciences (SMEMAG)

Title: Virtual charts for earthquake engineering including loading parameters

Keywords: Large Time Increment method for nonlinear low-frequency dynamics, Proper Generalized Decomposition, Time-Discontinuous Galerkin Method, Time-multiscale signal decomposition, Hyper-reduction method, Parallel strategy in time.

Abstract: The complexity and the fineness of the numerical models used to predict the (often nonlinear) seismic behavior of reinforced concrete structures impose a calculation time of several days to solve the partial differential equations of the reference problem. Moreover, for margin assessment, safety analyses or model updating purposes, taking into account the uncertainties associated with constitutive parameters or the seismic loading itself generally impose to foresee this numerical effort, not for the simulation of a single model, but of a family of models submitted to a set of likely inputs defining the seismic scenario. To reduce computational times, certain techniques, called “model order reduction", must be considered. The Large Time Increment (LATIN) method used in combination with the model order reduction technique called Proper Generalized Decomposition (PGD) has proven its efficiency for solving nonlinear problems in mechanics. Until now, the LATIN-PGD methodology has never been applied for solving nonlinear problems in dynamics. In this context, the LATIN-PGD framework is first adapted to the dynamic case, where the nonlinearities considered correspond to typical reinforced concrete materials, i.e. quasi-brittle isotropic damage for concrete and elasto-visco-plasticity for metals; additionally, dedicated strategies are developed to reduce computational costs when considering complex and large duration excitations, such as seismic inputs or fatigue loading. The contributions of this thesis work are the following: (i) an adaptation of the Time-Discontinuous Galerkin Method (TDGM) for solving incremental temporal problems in the LATIN-PGD framework, which allows to efficiently solve problems where the time interval is relatively large, (ii) a new signal approximation technique and a new multiscale strategy in time is developed to optimize the resolution of temporal PGD functions when dealing with long time excitations such as seismic inputs or fatigue loading, (iii) a hyper-reduction technique is proposed to accelerate the construction of the low-rank PGD approximation and finally, (iv) a parallel-time resolution strategy based on the use of TDGM is introduced, which aims at accelerating the temporal PGD resolution by taking advantage of the parallel architectures of recent computers. All the previous contributions allow a high optimization of the LATIN-PGD framework, which consequently allows a considerable reduction of the numerical cost to obtain the nonlinear response of a structure in dynamics.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l’Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France

To my parents, who with their great effort, sacrifice and dedication for years, formed the person I am today. This work is as much theirs as mine.

Acknowledgements

It has been a period full of learning, both professionally and personally, a unique and wonderful experience!

First of all, I would like to deeply thank each member of the jury for having the kindness to accept to be part of the evaluation team of my thesis. A special thanks to the president of the jury David Dureisseix for his very pertinent and detailed corrections that allowed me to improve the quality of the manuscript. Additionally, I would like to thank Elias Cueto and Delphine Brancherie, both reporters of my thesis, who devoted their valuable time to read my work in detail, providing precise observations that allowed me to better prepare my thesis defence. I would also like to deeply thank Andrea Barbarulo and Giuseppe Abbiati for being examiners of my thesis, with special thanks to Andrea, who was one of my professors at CentraleSupélec (in my academic exchange period back in 2015) and who, seeing my enthusiasm for learning and discovering science, helped and motivated me from the very beginning to do a PhD thesis. His quality as a person and his unconditional help has allowed me to become a doctor today!.

I would also like to thank the supervising team of this thesis, starting with David Néron for directing this thesis and allowing me to enter and discover the exciting area of reduced models and non-linear resolution using the LATIN method. Thank you for giving me your confidence and time in spite of your commitments. I also thank with all my heart Pierre Ladevèze, who with his characteristic patience was able to teach me how to do research, how to produce ideas, a true genius with whom I have had the privilege to work. I also thank Georges Nahas for his motivation and interest in always supporting my thesis work. Finally, I thank Pierre-Etienne Charbonnel, who more than a thesis supervisor is a great friend, a person who was always ready to help, his sympathy and constant support have made this thesis period much easier, he is a great professional and an even better person!.

I cannot forget in this incredible gratitude speech the family of doctoral students at the Ecole Normale Supérieure (ENS) Paris-Saclay, with whom I shared countless chats in front of the coffee machine at the old campus in Cachan, and at the new campus in Gif-sur-Yvette.Doing my thesis at the ENS Paris-Saclay was an adventure, meeting wonderful people, who made my thesis years a unique experience. Among these people I would like to mention the help and friendship of the remarkable Philippe (i.e. Philipo), the funny chats with Amelie, my amigo Flo, Nico Chan, Pierre-Jean (quelle vie), Francesco Ricardi, Flavian, Justin, Ariane, Alexandre, Matthieu, the wonderful girls from Lebanon Pascale and Aya, as well as the Italian team Livio and Marchelo! In addition to all the people mentioned above I would like to thank everyone I met at ENS Paris-Saclay, many of whom I don't mention here but who know who they are and how important they were during this time.

Finally, I would like to thank the most important people, who without their sacrifice for many years I would not even have the opportunity to be writing these words today. These people are my father (Jorge) and my mother (Rosa), who sacrificed for many years to give me and my siblings the best education possible. Additionally I would like to thank my brothers Jorge and Gustavo who are always there to support me.

Contents

1 Introduction **1**

2 Context and reference problem **6**

2.1 Introduction . . . 7

2.2 Reference problem . . . 7

2.2.1 Elastic formulation in dynamics . . . 8

2.2.1.1 Strong formulation . . . 8

2.2.2 Weak formulation . . . 9

2.2.2.1 Space weak formulation . . . 9

2.2.2.2 Space - time weak formulation . . . 9

2.2.3 Discretization of the space-time problem . . . 10

2.2.3.1 Space approximation . . . 10

2.2.3.2 Temporal approximation . . . 11

2.2.4 Nonlinear solid mechanics in dynamics . . . 14

2.3 Nonlinear constitutive relations considered . . . 16

2.3.1 Isotropic damage model for concrete materials . . . 16

2.3.1.1 Quasi-brittle behavior in tension . . . 17

2.3.1.2 Progressive micro-crack re-closure . . . 18

2.3.1.3 Synthesis . . . 19

2.3.1.4 Constants considered in the whole manuscript . . . 19

2.3.2 Elasto-visco-plastic constitutive relation . . . 21

2.3.2.1 Internal variables and thermodynamical framework . . . 21

2.3.2.2 Choice of the potential functions; state equations . . . 22

2.3.2.3 Evolution laws . . . 22

2.3.2.4 Normal formulation applied to the isotropic hardening . . . 23

2.3.2.5 Synthesis . . . 24

2.3.2.6 Constants considered in the whole manuscript . . . 24

2.4 Conclusions . . . 25

3 Overview of MOR techniques and their application for solving nonlinear problems 26

3.1 Introduction . . . 27

3.2 Low-rank approximation for time-dependent/parametric problems . . . 27

3.2.1 Approximation of rank m . . . 27

3.2.2 *A posteriori* Model Order Reduction - *Proper Orthogonal Decomposition* (POD) . . . 28

3.2.2.1 Best approximation of rank m . . . 29

3.2.2.2 Discrete *a posteriori* construction of the approximation . . . 30

3.2.3 Galerkin projection onto a reduced space-basis in linear dynamics problems . . . 31

3.2.3.1 Synthesis of the Proper Orthogonal Decomposition . . . 31

3.2.4 *Certified reduced basis method* (CRBM) . . . 32

3.2.4.1 General description for elliptic partial differential equations . . . 32

3.2.4.2 Reduced basis generation . . . 33

3.2.4.3 Synthesis of the method . . . 34

3.2.5 *A priori* Model Order Reduction - *Proper Generalized Decomposition* (PGD) . . . 35

3.2.5.1 Rank 1 approximation: Galerkin orthogonality criterion (G)PGD . . . 36

3.2.5.2 Rank 1 approximation: Minimization of the residual function (R)PGD . . . 36

3.2.5.3 Improvements to the Galerkin (G)PGD and the residual minimization (R)PGD strategies . . . 37

3.2.5.4 Different strategies for the construction of the decomposition . . . 37

3.2.5.5 Synthesis of the Proper Generalized Decomposition . . . 38

3.3 Incremental nonlinear solvers that use low-rank approximations . . . 40

3.3.1 Newton-Raphson method: framework for the incremental resolution of nonlinear solid mechanics problems . . . 40

3.3.1.1 Incremental resolution . . . 40

3.3.2 CRBM framework . . . 41

3.3.3 *A Priori Hyper-Reduction* (APHR) method . . . 42

3.4 Non incremental nonlinear solver that use low-rank approximations: the LATIN-PGD framework . . . 44

3.4.1 Introduction . . . 44

3.4.2 Resolution of parametrized nonlinear solid mechanics problems . . . 47

3.4.2.1 Case of elasto-visco-plastic problems . . . 47

3.4.2.2 Case of isotropic damage concrete problems . . . 49

3.4.3 Treatment of high-cycle fatigue . . . 50

3.4.4 Domain decomposition, parallel resolution and multiscale strategy . . . 50

3.4.5 Actual limitations of the method . . . 52

3.5 Conclusions . . . 52

4 The LATIN-PGD method extended to low-frequency dynamics problems 54
4.1 Introduction . . . 55
4.2 Initial elastic solution . . . 57
4.2.1 Approximation of the elastic dynamic solution . . . 58
4.3 Isotropic damage modeling for concrete material . . . 61
4.3.1 Local nonlinear stage . . . 61
4.3.2 Global linear stage: Equilibrium and compatibility equations . . . 62
4.3.2.1 Enrichment step: enrichment of the PGD approximation . . . 64
4.3.2.1.1 Space and time resolution associated to the admissibility problem 65
4.3.2.2 Preliminary step . . . 67
4.3.2.2.1 Update of the temporal functions . . . 68
4.3.3 LATIN error indicator . . . 70
4.4 Elasto-visco-plastic material . . . 71
4.4.1 Local nonlinear stage . . . 71
4.4.2 Global linear stage: equilibrium and compatibility equations . . . 73
4.4.2.1 Enrichment step: Enrichment of the PGD approximation . . . 75
4.4.2.1.1 Space and time resolution associated to the admissibility quantities 76
4.4.2.1.2 Space and time resolution associated to the internal variables . . 81
4.4.2.2 Preliminary step . . . 83
4.4.2.2.1 Time functions associated to the admissibility quantities . . . 83
4.4.2.2.2 Time functions associated to internal variables . . . 85
4.4.3 LATIN error indicator . . . 86
4.5 Synthesis on the LATIN-PGD solver . . . 86
4.6 Numerical examples . . . 88
4.6.1 Isotropic damage example . . . 90
4.6.2 Elasto-visco-plasticity example . . . 92
4.6.3 Comparison between the discontinuous (TDGM) and continuous (TCGM) approaches for the temporal resolution in the LATIN-PGD method . . . 95
4.6.4 Conclusions on the numerical results . . . 97
4.7 Conclusions . . . 98

5 LATIN-PGD multiscale in time for the resolution of complex fatigue problems 99
5.1 Introduction . . . 100
5.1.1 Previous multiscale approximation for simple fatigue problems . . . 100
5.1.2 New temporal multiscale approximation . . . 102
5.1.3 Important remarks . . . 106
5.1.3.1 Initial elastic solution . . . 106

5.1.3.2 Local stage . . . 106
5.2 Novel temporal multiscale approximation of signals . . . 107
5.2.1 Determination of the multiscale sub-modes . . . 107
5.2.1.1 "Greedy" calculation of multiscale sub-modes . . . 107
5.2.1.1.1 Determination of the characteristic periods . . . 109
5.2.1.1.2 Considerations of the macro discretization and resolution . . . 109
5.2.1.2 Error calculation . . . 110
5.2.2 Synthesis of the signal approximation . . . 110
5.2.3 Numerical examples: approximation of seismic signals . . . 111
5.3 New temporal multiscale approach applied to the LATIN-PGD solver . . . 114
5.3.1 Temporal multiscale approach applied to isotropic damage . . . 116
5.3.1.1 Enrichment step: computation of the temporal PGD functions . . . 117
5.3.1.2 Preliminary step: updating of the time functions . . . 121
5.3.2 Temporal multiscale approach applied to elasto-visco-plasticity . . . 124
5.3.2.1 Enrichment step: computation of the temporal PGD functions . . . 125
5.3.2.1.1 Temporal PGD function associated to the admissibility problem . . . 125
5.3.2.1.2 Internal variables . . . 128
5.3.2.2 Preliminary step: updating of the time functions . . . 129
5.4 Numerical examples . . . 129
5.4.1 Isotropic damage example . . . 130
5.4.2 Elasto-visco-plasticity example . . . 135
5.4.3 Conclusions on the numerical results . . . 139
5.5 Conclusions . . . 140

6 Hyper-reduction technique applied to the LATIN-PGD method 142
6.1 Introduction . . . 143
6.2 Hyper-reduction technique for function approximation . . . 145
6.2.1 Reference points in time: determination of the spatial POD functions . . . 145
6.2.2 Reference points in space: determination of the temporal POD functions . . . 147
6.2.3 Synthesis of the strategy . . . 148
6.2.4 Example: 2D space-time function approximation . . . 149
6.3 Hyper-reduced LATIN-PGD applied to elasto-visco-plasticity . . . 150
6.3.1 Local stage: hyper-reduced evaluation of the constitutive relations . . . 151
6.3.1.1 Reference points in time: determination of the spatial POD functions . . . 153
6.3.1.2 Reference points in space: determination of the temporal POD functions . . . 154
6.3.1.3 Extensibility of the low-rank decomposition defined on the reduced domain Ω_{nl} to the total space domain Ω . . . 155

6.3.1.4 Rank determination of the decomposition . . . 155
6.3.1.5 Error estimator of the approximation . . . 156
6.3.2 Global linear stage: Equilibrium and compatibility equations . . . 157
6.4 Numerical example . . . 159
6.5 Conclusions . . . 161

7 Parallel strategy in time applied to the LATIN-PGD method 163
7.1 Introduction . . . 164
7.2 Initial elastic problem . . . 166
7.3 Local stage . . . 166
7.4 Global stage: Parallel resolution of the temporal functions . . . 167
7.4.1 Enrichment step: temporal function determination . . . 169
7.4.1.1 Parallel resolution . . . 171
7.4.2 Preliminary step: Temporal functions actualization . . . 173
7.5 Numerical example . . . 175
7.6 Conclusions . . . 178

8 Conclusions and perspectives 179

9 Appendix 182

A Bilinear and linear discretization of the elastic dynamic problem 183

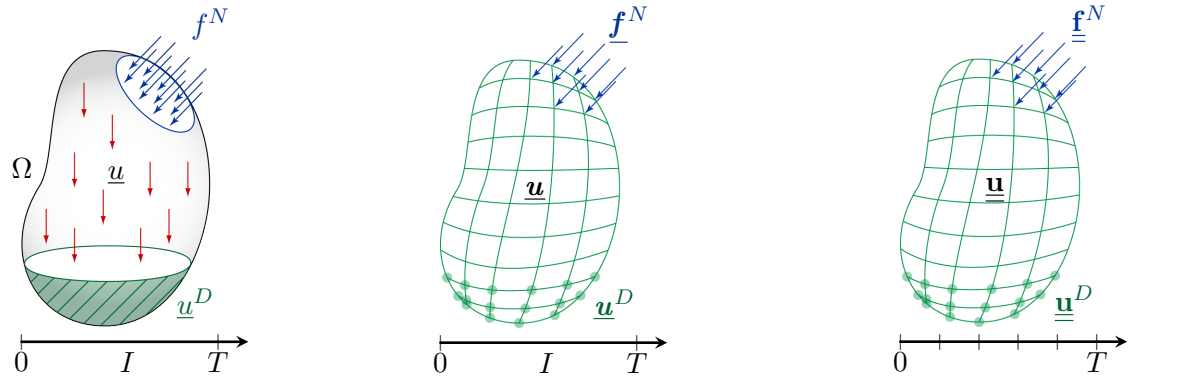
B General framework for the incremental resolution of nonlinear solid mechanics problems 186
B.1 Temporal incremental approximation . . . 186
B.2 Incremental resolution . . . 187
B.2.1 Spatial discretization and temporal incremental resolution . . . 188

C Enrichment step: temporal problem of internal variables by using TDGM 190

D Temporal multiscale approximation applied to the preliminary step when dealing with elasto-visco-plasticity 192
D.1 Temporal functions associated to the admissibility problem . . . 192
D.2 Internal variables . . . 196

Notations

(a) Continuous problem ***(b) Space discretization*** ***(c) Temporal discretization***



Solution $\underline{u}(\underline{x},t)$ $\quad \underline{u}^h(\underline{x},t) = \chi(\underline{x}).\underline{\boldsymbol{u}}(t)$ $\quad \underline{u}^{\mathcal{N}}(\underline{x},t) = \left[\chi(\underline{x}) \otimes \underline{\psi}(t)\right] : \underline{\underline{\mathbf{u}}}$

For all the thesis the following convection is adopted:

- The **continuous** tensors of order 1,2 and 4 are noted as $\underline{u}$, $\underline{\sigma}$ and $\mathbb{K}$ respectively (fourth order tensor is written in a simplified form to alleviate notations), for example the Hooke's law is written as:

$$\underline{\sigma} = \mathbb{K} : \underline{\varepsilon}$$

- The **semi-discretized** vectors and matrices are denoted as $\underline{\boldsymbol{u}}$ and $\underline{\underline{\boldsymbol{M}}}$ respectively, for example, the semi-discretized dynamic equation is written as:

$$\underline{\underline{\boldsymbol{M}}}\,\underline{\ddot{\boldsymbol{u}}}(t) + \underline{\underline{\boldsymbol{D}}}\,\underline{\dot{\boldsymbol{u}}}(t) + \underline{\underline{\boldsymbol{K}}}\,\underline{\boldsymbol{u}}(t) = \underline{\boldsymbol{f}}(t)$$

- The **discretized** matrices in space and time are denoted in bold $\underline{\underline{\mathbf{u}}}$, for example, the discretized dynamic equation writes:

$$\underline{\underline{\underline{\mathbf{A}}}} : \underline{\underline{\mathbf{u}}} = \underline{\underline{\mathbf{B}}}$$

with $\underline{\underline{\underline{\mathbf{A}}}} \in \mathbb{R}^{n_S \times n_S} \otimes \mathbb{R}^{n_T \times n_T}$ and $B \in \mathbb{R}^{n_S \times n_T}$, and where we considered the following operation:

$$(\underline{\underline{\boldsymbol{A}}} \otimes \underline{\underline{\boldsymbol{B}}}) : \underline{\underline{\boldsymbol{C}}} = \underline{\underline{\boldsymbol{D}}} \Longleftrightarrow \sum_{j,l} A_{ij} B_{kl} C_{jl} = D_{ik}$$

Settings of the reference problem

$\underline{x}$	Position vector.
t	Time.
$\Omega \subset \mathbb{R}^d$	Spatial domain.
$I = [0, T]$	Temporal domain.
$I_\theta \subset \mathbb{R}^P$	Parametric space.
$d \in [1, 2, 3]$	Dimension of the space domain.

Operations

$\cdot$	Canonical scalar product between two vectors or fields.
$:$	Canonical scalar product between tensors of order 2.
$\overset{\mathsf{D}}{\cdot}$	Canonical scalar product between tensors of order D (D equals 1 or 2).
$\otimes$	Tensor product between tensors, matrices, functions or spaces.

Norms

$\|\cdot\|_2$	Euclidean norm.
$\|\cdot\|_\Omega^2 = \int_\Omega (\cdot) \overset{\mathsf{D}}{\cdot} (\cdot)\, d\Omega$	canonical spatial norm.
$\|\cdot\|_I^2 = \int_I (\cdot) \overset{\mathsf{D}}{\cdot} (\cdot)\, dt$	canonical temporal norm.
$\|\|\|\cdot\|\|\| = \int_{\Omega\times I} (\cdot) \overset{\mathsf{D}}{\cdot} (\cdot)\, d\Omega dt$	canonical norm over the space and time.
$\|\|\|\cdot\|\|\|_{\mathbb{H}}^2 = \int_{\Omega\times I} (\cdot) : \mathbb{H} : (\cdot)\, d\Omega dt$	tensor norm over the space and time.

Space of functions

$L(E, F)$	Space of linear applications from E to F.
$L^2(E, F)$	Space function from E to F that are square integrable.
$\mathcal{H}_1(E, F)$	Sobolev space of functions from E to F.

Solutions and space solutions

$\underline{u}(\underline{x}, t) \in \mathcal{U}$	Displacement field, solution of the space-time problem.
$\mathcal{U}^S \subset \mathcal{H}_1(\Omega, \mathbb{R}^d)$	Solution space of the spatial problem with fixed time t, $\underline{u}(\underline{x}, t)$.
$\mathcal{U}^T \subset L^2(I, \mathbb{R})$	Space of solution $\underline{u}_i(\underline{x}, t)$ by component of the temporal problem at fixed $\underline{x}$.
$\mathcal{U} \subset L^2(I, \mathcal{U}^S)$	Space solutions of the finite-energy space-time problem of I in $\mathcal{U}^S$.
$\underline{\chi}(\underline{x}) \in [\chi_1, \ldots, \chi_{n_S}]$	Basis of n_S shape functions associated to the space discretization.
$\mathcal{U}_h^S = \mathsf{span}\{\chi_i\}_{i=1}^{n_S}$	Space of the spatial approximation of $\mathcal{U}^S$ of dimension n_S.
$\underline{\boldsymbol{u}}(t) \in \mathbb{R}^{n_S} \otimes \mathcal{U}^T$	Semi-discretized displacement field.
$\underline{u}^h(\underline{x}, t) \in \mathcal{U}_h^S \otimes \mathcal{U}^T$	Finite element approximation (semi-discretized) of the solution $\underline{u}$.
$\underline{\psi}(t) \in [\psi_1, \ldots, \psi_{n_T}]$	Basis of n_T shape functions associated to the temporal discretization.
$\mathcal{U}_{\Delta t}^T = \mathsf{span}\{\psi_i\}_{i=1}^{n_T}$	Temporal approximation space of $\mathcal{U}^T$ of finite dimension n_T.
$\underline{\underline{\mathbf{u}}} \in \mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T}$	Matrix containing the unknown space-time nodal values.
$\underline{u}^{\mathcal{N}}(\underline{x}, t) \in \mathcal{U}_h^S \otimes \mathcal{U}_{\Delta t}^T$	Finite element approximation in space and time of the solution $\underline{u}$.

Terminology / Abbreviations

FEM	*Finite Element Method.*
CGM	*Continuous Galerkin Method.*
DGM	*Discontinuous Galerkin Method.*
POD	*Proper Orthogonal Decomposition.*
PCA	*Principal Component Analysis.*
SVD	*Singular Value Decomposition.*
EVD	*Eigen Value Decomposition.*
CRBM	*Certified Reduced Basis Method.*
EIM	*Empirical Interpolation Method.*
LATIN	*LArge Time INcrement method.*
PGD	*Proper Generalized Decomposition.*
FT	*Fourier Transformation.*

Chapter 1

Introduction

The problems raised by *Seismic Risk Assessment* and *Earthquake Resistant Design* are subjected to a high degree of uncertainties. The major lack-of-knowledge resides in the excitation itself that, by nature, can not be characterized in a deterministic manner. In most cases, the likely seismic scenario is described by a vector θ of uncertain parameters (magnitude, site-to-source distance, local shear wave velocity, etc.) that usually lives in a broad probabilistic space I_θ. Even if the seismic scenario was precisely specified, it would of course not lead to a unique representative seismic action to apply to structures. The literature for generating potential seismic inputs from seismic scenario $\theta \in I_\theta$ is vast [Douglas and Aochi, 2008, Charbonnel, 2018] and can be based on regression techniques on complete seismic database (see e.g. NGA-PEER Database [Power et al., 2008, Chiou and Youngs, 2008, Baker and Jayaram, 2008, Boore and Atkinson, 2008, Abrahamson and Silva, 2008] in California or RESORCE Database [Akkar et al., 2014] in Europe), on single recordings or spectral specifications (see e.g. [Rezaeian and Der Kiureghian, 2008, Rezaeian and Der Kiureghian, 2010, Yamamoto and Baker, 2011, Lancieri et al., 2012, Zentner et al., 2013, Rossetto et al., 2016, Lancieri et al., 2018]) or on full physical soil medium modeling (see e.g. [Zerva, 1988, Gatti et al., 2018]). To account for the huge variability of the possible ground motion inputs and according to the different earthquake resistant design recommendations for civil constructions [EUROCODE-8, 2004, Elghazouli, 2009, NEHRP, 2010, Bisch et al., 2012] numerical models, together with their own uncertainties (material parameters), must be subjected, not only to a single seismic loading, but to a set of potential seismic inputs (see figure 1.1).

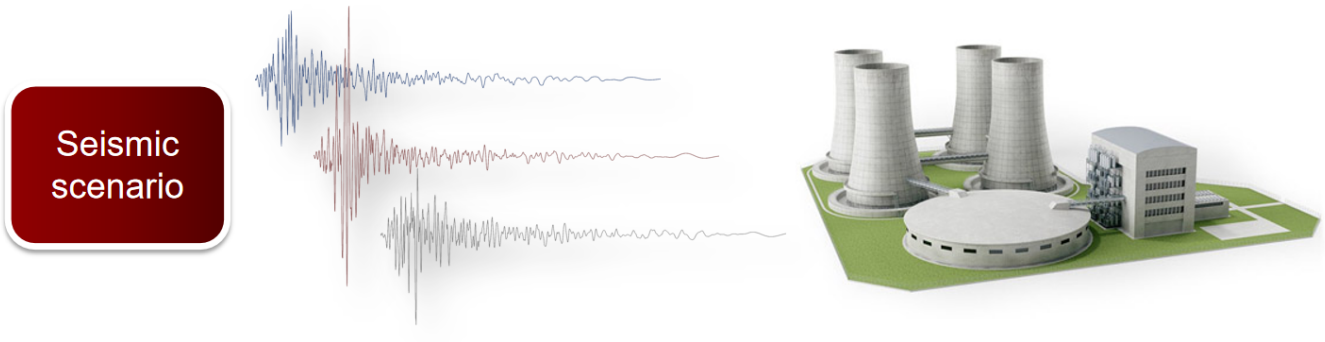


Figure 1.1: Samples of potential ground motions (full scales harmonized) generated from a single acceleration recording using the methodology proposed by [Rezaeian and Der Kiureghian, 2010].

Of course *Nuclear Power Plants* (NPPs) qualification does not except to this rule [IAEA, 2003]. Moreover after the Fukushima Dai-ichi accident, safety requirements have tightened; two consecutive European projects, leaded by french nuclear actors, were launched with the objective of assessing the

NPPs robustness against extreme events and identifying whether some reinforcements were needed. The ASAMPSA_E (2013–2016) project first aimed at promoting good practices to extend the scope of existing probabilistic safety assessments in decision-making. This project led to a collection of guidance reports that described existing practices and identified their limits. The current H2020 project NARSIS (2017–2021) in turn aims at proposing some improvements to be integrated in existing probabilistic safety assessment procedures for NPPs, considering single, cascade and combined external natural hazards (earthquakes, flooding, extreme weather, tsunamis). Overview of both projects and further details are given in [Foerster et al., 2020]. The project will lead to the release of various tools together with recommendations and guidelines for use in nuclear safety assessment.

The present doctoral work falls within the scope of seismic risk assessment and focuses on developing advanced numerical framework for obtaining decision-making tools, such as fragility curves, using high fidelity Finite Element Models *(FEMs), with the objective of decreasing associated computational costs.*

For correctly answering the problematic, the methodology to be developed will have to match particular requirements, namely:

(R1) being suitable for low-frequency nonlinear dynamics,

(R2) capable of handling complex loadings,

(R3) adapted to typical reinforced-concrete nonlinearities due to the major role it plays in earthquake engineering applications.

Among the different strategies dedicated to reduce the computational efforts, some methods released in the 2000s are currently booming. They propose to use an ingredient referred to as *model order reduction* (MOR) which confers them a powerful numerical efficiency. The main idea is to exploit the redundancy of information contained in the solution to propose an approximated and numerically efficient resolution of the problem, which guarantees that the calculated approximation, called low-rank approximation, stays close enough to the reference solution. For that purpose, the solution of the reference problem is thus approached by a sum of m terms where each of the terms is a product of functions with separate variables (space, time and eventually parameters). The integer m is called the rank of the approximation and in practice, the approximation space (which basis contains the separate variables functions) is constructed incrementally.

The model reduction techniques that are widely used correspond to the ***Proper Orthogonal Decomposition*** (POD) and the ***Proper Generalized Decomposition*** (PGD). The main difference between them lies on the way the low-rank approximation is calculated. While in the POD a data set of the problem solution is required for the construction of the approximation in a training phase, for PGD this is not necessary, the approximation is constructed on the fly using the partial differential equations of the problem. Due to the aforementioned differences in the way the low-rank approximation is constructed, the POD is referred to as *a posteriori* while the PGD is classified as *a priori* model order reduction.

When solving a given problem a POD reduced basis can be used. Since its introduction in [Karhunen, 1946, Loeve, 1948], the general idea of the method that uses the POD in its formulation consists in the "offline" construction of a reduced order basis (ROB) by using the full finite element solution of the reference problem in some specific points of the time or parameter domain. The reduced order basis is formed by considering only the most energetic POD modes. Once the reduced order model is built, it can be used in a "online" stage to achieve a high decrease of the computational cost when solving the reference problem over the whole time-parameter domain by projecting the discretized reference equations of the problem over the ROB, reducing in this way the total degrees of freedom of the problem. Among the uses of the POD we can cite its application to aeroelastic problems [Lieu et al., 2005, Lieu and Farhat,

2005, Lieu and Farhat, 2007, Amsallem and Farhat, 2008], computational fluid dynamics [Kunisch and Volkwein, 2002, Amsallem and Farhat, 2014], nonlinear dynamics [Kerschen et al., 2005], face image reconstruction [Everson and Sirovich, 1995], among many others.

A well known solver that exploits the POD approximation to solve nonlinear problems in solid mechanics is the *A Priori Hyper-Reduction* method. In this method, the POD basis used to approximate the solution is constructed "online" using an adaptive strategy and the reference problem is solved in a "Reduced Integration Domain" (RID), which allows to accelerate the solution of the nonlinear problem [Ryckelynck, 2005, Ryckelynck et al., 2006a]. The use of a reduced integration domain for the solution of the problem gives its name of hyper-reduction. This method has been successfully applied to solve nonlinear internal variable problems [Ryckelynck, 2009], damage problems [Ryckelynck et al., 2011] or parametric nonlinear problems [Ryckelynck et al., 2012].

Another important technique in the field of *a posteriori* model reduction is the *Certified Reduced Basis Method* (CRBM). The CRBM has been introduced initially in [Maday et al., 2002, Prud'Homme et al., 2002] for the resolution of elliptic parametric problems. The methodology resides in the sequential construction of reduced basis of the solution space on which governing equations are projected, which enables the accelerated "online" resolution of the parametric problem. The reduced basis itself is constructed "offline" and incrementally by determining the reduced basis as a set of reference solutions over a given parametric set. This set is efficiently determined by employing error estimators which guarantees an optimal selection of the set of parametric solutions that minimize a reconstruction error over the whole parametric domain. One of the underlying hypotheses for the "online" computation of the parametric solution is the affine decomposition of the operators related to the reference problem with respect to the parameter vector; i.e. the linear and bilinear forms involved in the problem definition must be written as a sum of products of contributions with separated space-parameter variables. This so-called affine decomposition allows the construction of the spatial operators only once, allowing enormous computational savings during the online resolution. The CRBM has then been adapted for parabolic problems and extended in the nonlinear range for the resolution of problems having any kind of parametric dependence by employing the *Empirical Interpolation Method* (EIM) [Barrault et al., 2004, Grepl et al., 2007, Maday et al., 2009] for the approximation of non affine operators as an affine sequence (see [Prud'Homme et al., 2002, Grepl, 2005, Quarteroni et al., 2011, Veys, 2014] for details).

Alternatively, the PGD was introduced in [Ladevèze, 1985] under the vocable "radial approximation" as one of the main ingredients of the LATIN method. The LATIN method, for *LArge Time INcrement* method, proposes a general strategy for the resolution of nonlinear problems in mechanics involving an alternating sequence of nonlinear and linear steps. At each linear step, a global space-time problem expressing the equilibrium of the system must be solved and the PGD is used to provide a reliable and numerically economical low-rank approximation of the solution of this linear problem.

Since its introduction in [Ladevèze, 1985], the PGD has been the subject of numerous other publications for the resolution of different types of linear problems. For instance, parametric linear problems are solved following both deterministic [Ammar et al., 2006, Chinesta et al., 2010, Ammar et al., 2010, Ammar et al., 2012] and stochastic [Nouy, 2008, Nouy, 2009, Nouy, 2010b] approaches where rather than PGD, the denomination Generalized Spectral Decomposition (GSD) is used. Other developments intend to improve its performance for time dependent solid mechanics problems as in [Nouy, 2010a] and [Boucinha et al., 2013, Boucinha et al., 2014]. In addition, a wide number of applications of the PGD have shown its performance by reducing the computational cost for many different kind of problems, for instance mid-frequency problems in the context of the *Variational Theory of Complex Rays* (VTCR) [Barbarulo, 2012], and further developed in [Chevreuil and Nouy, 2012, Barbarulo et al., 2013, Riou et al., 2013, Barbarulo et al., 2014a, Barbarulo et al., 2014c, Barbarulo et al., 2014b], hyperelastic materials [Niroomandi et al., 2010], correlation of high resolution digital images [Passieux and Périé, 2012], parameterized Helmholtz problem [Signorini et al., 2017], geometrically parameterized heat problem [Zlotnik et al., 2015], among a

whole range of different applications.

This doctoral work proposes several enhancements for applying the LATIN-PGD methodology in nonlinear low-frequency dynamics for the purpose of seismic risk assessment. More precisely, for evolutionary problems, the LATIN approach consists in a sequence of (i) *local stages* where constitutive relations are solved for each Gauss point of the finite element formulation, and (ii) *global stages* where the equilibrium on the whole space-time domain is imposed. The most numerically expensive task is by far the re-imposition of the global equilibrium at the second step (ii) ; however, this step is linear and its solution can be approached using a low-rank approximation, known as the *Proper Generalized Decomposition* (PGD), which greatly improves the numerical performance of the approach. The LATIN-PGD method has proved particularly efficient for treating nonlinear problems such as elasto-visco-plasticity in [Ladevèze et al., 2010b, Relun et al., 2013], a multiscale approach for the simulation of composite materials [Passieux, 2008, Relun et al., 2011]. Large parametric problems for materials under quasi-static conditions and with typical elastic-visco-plastic behavior have been treated in [Boucard and Ladevèze, 1999, Néron et al., 2015] or more recently [Vitse et al., 2019] on reinforced concrete. Also an hyper-reduction technique called *Reference Point Method* specially dedicated for the approximation of the tangent operator of the LATIN-PGD in elasto-visco-plasticity was developed in [Capaldo et al., 2017].

In the parametric nonlinear context, the PGD within the LATIN framework was successfully used for dealing with a wide range of problems including, heat transfer [Heyberger et al., 2012, Heyberger et al., 2013], the study of solid mechanics problems considering as nonlinearity elasto-visco-plasticity [Relun et al., 2013, Néron et al., 2015], and more recently reinforced concrete under quasi-static conditions [Vitse et al., 2019], where material properties and amplitude load have been parameterized. The efficiency of the PGD made it possible to deal with problems involving up to a dozen of parameters but passed that number, the PGD seem to loose its empirically observed convergence properties. Recent developments [Paillet et al., 2017, Ladevèze et al., 2018, Paillet et al., 2018a, Paillet et al., 2018b], based on the Saint-Venant's principle and a multiscale description of the parameter dependency, propose to overcome this limitation and extend the scope of the method to the case of high dimensions.

Finally, due to it is main characteristic, i.e. the global resolution of the problem in space and time at the linear stage, the idea of treating cyclic fatigue problems on large time intervals naturally arose. Many works have been done in this domain, the first one was introduced by [Cognard and Ladevèze, 1993] for the case of viscoplasticity and, a year later, [Arzt and Ladevèze, 1994] extended some of the notions of the first work. More recent developments for the case of viscoplasticity and damage for fatigue problems were presented on [Bhattacharyya et al., 2018a, Bhattacharyya et al., 2018b, Bhattacharyya et al., 2018c] and [Bhattacharyya et al., 2019]. The main idea introduced in the aforementioned references is, from the observation that the evolution of the solution is very slow according to loading time, to compute the time corrections on so-called *nodal cycles*, the rest of the solution being interpolated.

The LATIN-PGD alliance strategy then represents an efficient framework for nonlinear problems, but from now on, its range of applications only assumes quasi-static evolution. The studies proposed by earthquake engineering applications impose to bypass this limitation. The aim of the present work is to extent the framework of the LATIN-PGD methodology in order to simulate nonlinear dynamics problems in mechanics, and to develop dedicated strategies for the fast resolution of the temporal functions related to the PGD decomposition when dealing with complex excitations of long duration in time.

The contributions of this doctoral work to address the seismic risk assessment problematic and match requirements *(R1)*, *(R2)* and *(R3)* can be summarized as below:

1. Extension of the LATIN-PGD framework to the case of nonlinear vibratoric dynamic problems in solid mechanics.

2. Implementation of a new multiscale signal approximation, together with a new multiscale strategy for

solving the temporal PGD functions in the global stage of the LATIN method, which seeks to optimize the resolution of nonlinear problems when considering complex input signals of long duration in time, such as seismic excitations or fatigue loads.

3. Application of a hyper-reduction technique for the reduction of the computational cost associated with the evaluation of the constitutive relation in the local stage, which also approximates the solution of the local stage (local variables) under a low-rank decomposition that optimizes the spatio-temporal integration operations required in the global stage of the LATIN-PGD method. Optimizing the solver from a numerical point of view when solving problems of long time duration.

4. Definition of a parallel resolution strategy in time in order to bring additional efficiency when solving problems where the imposed excitation have a long duration and a complex behavior in time.

The present manuscript is then structured as follows:

- **Chapter 2** introduces the mathematical formulation of the reference problem at the core of our study along with the constitutive relations considered in this work. The developments detailed in next chapters will refer to the two constitutive relations associated to elasto-visco-plasticity typical of steel reinforcements and isotropic damage for quasi-brittle material such as concrete medium. Please note that the two constitutive relations are considered independently; no reinforced-concrete modeling (both models working in parallel) is developed in this work.

- **Chapter 3** gives a concise state-of-the-art regarding model order reduction techniques as well as their usage for solving linear and nonlinear solid mechanics problems. The main ingredients of the LATIN-PGD solver are explained at the end of this chapter.

- **Chapter 4** describes in detail the LATIN-PGD methodology used to solve low-frequency dynamics problems. This chapter uses the two constitutive relations presented in chapter 2 to explain the methodology and implementation details for each particular case.

- **Chapter 5** presents the new temporal multiscale strategy for signal approximation and the multiscale approximation of the temporal PGD functions of the LATIN solver for the two constitutive relations considered.

- **Chapter 6** details the hyper-reduction technique applied to the local stage of the LATIN-PGD method for the fast evaluation of the constitutive relation quantities and its low-rank approximation for the optimization of the PGD resolution in the global stage when dealing with long time duration excitations.

- **Chapter 7** presents a parallel strategy based on the use of Time Discontinuous Galerkin Method (TDGM), specifically dedicated to the calculation of the temporal PGD functions of the LATIN-PGD, this in order to give a new way of reducing the computational expenses by exploiting a parallel hardware architecture.

All presented developments have been developed in the Matlab framework. This work is part of the WP.4 work package of the NARSIS project that is acknowledged for partially supporting this doctoral work. The tandem CEA-IRSN is also deepfully thanked for funding this research activity.

Chapter 2

Context and reference problem

This chapter introduces the notations of the reference problem in dynamics considered throughout this doctoral work. Strong and weak forms of the reference problem are recalled. Both continuous and discretized spaces are presented along with the Finite Element Method (FEM) for its approximation. Finally, the two nonlinear constitutive relations considered in this work and involved in the derivation of the methodology in the next chapters are also detailed.

Contents

2.1 Introduction **7**
2.2 Reference problem **7**
2.3 Nonlinear constitutive relations considered **16**
2.4 Conclusions **25**

2.1 Introduction

As explained in chapter 1, the main objective of this thesis is to efficiently solve nonlinear solid mechanics problems in low-frequency dynamics. For this purpose, it is necessary to first formalize the reference problem by formulating it mathematically. Once the mathematical foundations have been presented, it is possible to move easily towards the introduction of methodologies that allow the efficient solution of this problem.

This chapter is structured as follows: first, the mathematical bases of an elastic problem in dynamics are presented, where a global spatio-temporal formulation is considered because of its use in future chapters. Secondly, the nonlinear problem in dynamics is introduced, considering as the source of nonlinearity the material behavior. A damaging medium is considered for modeling the concrete matrix and an elasto-visco-plastic behavior is considered for modeling steel reinforcements. The two constitutive relations at stake are detailed in the sections 2.3.1 and 2.3.2 respectively. Let us stress once again that the two constitutive relations are considered independently. The modeling of reinforced-concrete (both models working in parallel) is not carried out in this work.

Finally, the numerical values associated with the different parameters for each material are listed; these values are fixed and are used throughout this manuscript for each section of numerical results.

2.2 Reference problem

The reference problem concerns the dynamic isothermal evolution of a structure assuming small perturbations. Lets consider a structure occupying the domain $\Omega \in \mathbb{R}^d$ with $d \in \{1, 2, 3\}$ over the time domain $I = [0, T]$, with a boundary $\partial\Omega = \partial_N\Omega \oplus \partial_D\Omega$, where $\partial_N\Omega$ and $\partial_D\Omega$ are the boundaries related to the imposed Neumann and Dirichlet conditions respectively. This structure is subjected to external forces $\underline{f}^N$ in $\partial_N\Omega$, a displacements $\underline{u}^D$ on $\partial_D\Omega$ and to volumetric forces $\rho\underline{f}$ over Ω (with ρ the density of the material).

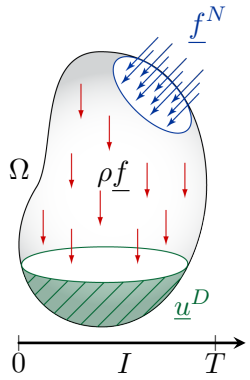


Figure 2.1: Continuous reference problem.

The unique source of nonlinearities taken into account in this thesis correspond to the material behavior; of course other nonlinearities (of geometric nature for example) could be taken into account, but this is out of the scope of the present work.

In what follows the mathematical formulations related to linear and nonlinear problems are presented. A global spatio-temporal formulation is presented for the linear problem, where concepts of great importance used throughout this manuscript are defined. After presenting the linear problem, a brief formulation is given for the nonlinear case.

2.2.1 Elastic formulation in dynamics

2.2.1.1 Strong formulation

The elastic reference problem consists in finding the displacement field $\underline{u}(\underline{x},t) \in \mathcal{U}$ and the stress field $\underline{\underline{\sigma}}(\underline{x},t) \in \mathcal{F}$ such that they verifies:

- Initial conditions on Ω:
$$\underline{u}\Big|_{t=0^+} = \underline{u}_{in} \qquad \text{and} \qquad \underline{\dot{u}}\Big|_{t=0^+} = \underline{\dot{u}}_{in} \tag{2.1}$$
Because we are dealing with vibratory dynamics problems, the initial conditions of equation (2.1) are considered to be null, otherwise we can produce a shock as initial condition that would produce propagative waves on the solid, a type of problem that is outside the scope of this thesis.

- Equilibrium equation over $\Omega \times I$:
$$\rho\underline{\ddot{u}} = \underline{\text{div}}(\underline{\underline{\sigma}}) + \rho\underline{f} \tag{2.2}$$
with $\underline{\text{div}}(\cdot)$ the divergence operator.

- Dirichlet and Neumann conditions:
$$\underline{\underline{\sigma}} \cdot \underline{n} = \underline{f}^N, \quad \text{over} \quad \partial_N\Omega \times I. \tag{2.3}$$
$$\underline{u} = \underline{u}^D, \quad \text{over} \quad \partial_D\Omega \times I. \tag{2.4}$$
with $\underline{n}$ the outward unit normal vector to the surface $\partial_N\Omega$.

- Linear elasticity constitutive relations:
When dealing with a linear problem the stress is simply given by the following elastic constitutive relation:
$$\underline{\underline{\sigma}}(\underline{u}) = \mathbb{K} : \underline{\underline{\varepsilon}}(\underline{u}) + \mathbb{D} : \underline{\underline{\varepsilon}}(\underline{\dot{u}}), \quad \text{over} \quad \Omega \times I. \tag{2.5}$$
where $\mathbb{K}$, $\mathbb{D}$, $\underline{\underline{\varepsilon}}$ and $\underline{\underline{\sigma}}$ respectively denote the Hooke tensor, the damping tensor, the elastic strain tensor and the elastic stress tensor. The elastic strain tensor is defined as:
$$\underline{\underline{\varepsilon}}(\underline{u}) = \frac{1}{2}\left(\nabla\underline{u} + \nabla\underline{u}^T\right) \tag{2.6}$$
where the $(\cdot)^T$ and $\nabla(\cdot)$ denotes the transpose and gradient operators respectively.

Remark: The Hooke tensor $\mathbb{K}$ for the elastic problem is considered to be isotropic (although it is not limited to this, anisotropic behaviour can also be considered), its matrix representation using Voigt notation is given as follows:

$$\begin{bmatrix}\sigma_{xx}\\ \sigma_{yy}\\ \sigma_{zz}\\ \sigma_{yz}\\ \sigma_{xz}\\ \sigma_{xy}\end{bmatrix} = \underbrace{\frac{E}{(1+\nu)(1-2\nu)}\begin{bmatrix}1-\nu & \nu & \nu & 0 & 0 & 0\\ \nu & 1-\nu & \nu & 0 & 0 & 0\\ \nu & \nu & 1-\nu & 0 & 0 & 0\\ 0 & 0 & 0 & \frac{1-2\nu}{2} & 0 & 0\\ 0 & 0 & 0 & 0 & \frac{1-2\nu}{2} & 0\\ 0 & 0 & 0 & 0 & 0 & \frac{1-2\nu}{2}\end{bmatrix}}_{\mathbb{K}}\begin{bmatrix}\varepsilon_{xx}\\ \varepsilon_{yy}\\ \varepsilon_{zz}\\ 2\varepsilon_{yz}\\ 2\varepsilon_{xz}\\ 2\varepsilon_{xy}\end{bmatrix} \tag{2.7}$$

where E and ν correspond to the Young modulus and the Poisson coefficient respectively.

2.2.2 Weak formulation

Taking advantage of the fact that the variables t and $\underline{x}$ play very different roles, we separate these variables by considering henceforth the solution $\underline{u}(\underline{x},t)$ as a function of time t with values in a function space defined in Ω (the same for the load $\underline{f}(\underline{x},t)$. We denote $\mathcal{U}^S$ this space of spatial functions. More precisely, in the domain $I=[0,T]$ we assume the solution $\underline{u}$ defined by:

$$\begin{array}{rl} \underline{u}: & I \to \mathcal{U}^S \\ & t \to \underline{u}(t) \end{array} \tag{2.8}$$

where $\underline{u}(t)$ is a function of the spatial variable $\underline{x}$. We will continue to note $\underline{u}(\underline{x},t)$ the value of the field $u(t)(\underline{x})$ evaluated at point $\underline{x}$.

2.2.2.1 Space weak formulation

The weak formulation of equation (2.2) in space consists in writing the local equation to a global one. Thus the solution field $\underline{u}$ is searched in the space $\mathcal{U}^S(\Omega,\underline{u}^D)$ of kinematically admissible functions defined as:

$$\mathcal{U}^S(\Omega;\underline{u}^D)=\left\{\underline{u}\in\mathcal{H}^1(\Omega,\mathbb{R}^d)\mid \underline{u}(\underline{x})=\underline{u}^D,\quad \forall \underline{x}\in\partial_D\Omega\right\} \tag{2.9}$$

where $\mathcal{H}^1(\Omega)$ defines the Sobolev space of the function $\underline{u}$ defined over Ω in $\mathbb{R}^d$, where $\underline{u}$ and $\nabla\underline{u}$ are square integrable. Once the above space defined, the weak formulation in space can be written:

$\forall \underline{v}\in\mathcal{U}^S(\Omega,0),\ \forall t\in I,$

$$m(\ddot{\underline{u}}(t),\underline{v})+d(\dot{\underline{u}}(t),\underline{v})+k(\underline{u}(t),\underline{v})=f(\underline{v};t) \tag{2.10}$$

with the initial conditions:

$\forall \underline{x}\in\Omega,$

$$\begin{array}{l} \underline{u}(\underline{x},0)=\underline{u}_{in}(\underline{x}) \\ \dot{\underline{u}}(\underline{x},0)=\dot{\underline{u}}_{in}(\underline{x}) \end{array} \tag{2.11}$$

The symmetric bilinear forms $m(.,.)$, $d(.,.)$, $k(.,.)$ are defined by:

$$m(\underline{u},\underline{v})=\int_\Omega \rho\underline{v}\cdot\underline{u}\,d\Omega \tag{2.12}$$

$$d(\underline{u},\underline{v})=\int_\Omega \underline{\underline{\varepsilon}}(\underline{v}):\mathbb{D}:\underline{\underline{\varepsilon}}(\underline{u})\,d\Omega \tag{2.13}$$

$$k(\underline{u},\underline{v})=\int_\Omega \underline{\underline{\varepsilon}}(\underline{v}):\mathbb{K}:\underline{\underline{\varepsilon}}(\underline{u})\,d\Omega \tag{2.14}$$

The linear form in turn writes:

$$f(\underline{v};t)=\int_\Omega \underline{v}\cdot\rho\underline{f}(t)\,d\Omega+\int_{\partial_N\Omega}\underline{v}\cdot\underline{f}^N(t)\,dS \tag{2.15}$$

2.2.2.2 Space - time weak formulation

As for the spatial case, a weak formulation in time can be written [Allaire, 2005], by reducing the space $\mathcal{U}$ where the solution is searched to the following space:

$$L^2(I;\mathcal{U}^S)=\left\{\underline{v}:I\to\mathcal{U}^S\ \middle|\ \int_I\|\underline{v}(t)\|_S^2\,dt<\infty\right\} \tag{2.16}$$

where $\|.\|_S$ is a norm over $\mathcal{U}^S$. By denoting $\mathcal{U}^T = L^2(I;\mathbb{R})$, the space $\mathcal{U} = L^2(I;\mathcal{U}^S)$ is identified to the tensor product space $\mathcal{U}^S \otimes \mathcal{U}^T$ [Allaire, 2005, Nouy, 2010a]. Thus a weak solution of the space-time problem consists in searching $\underline{u} \in \mathcal{U}^S(\Omega;\underline{u}^D) \otimes \mathcal{U}^T(I)$ verifying:

$\forall \underline{v} \in \mathcal{U}^S(\Omega;0) \otimes \mathcal{U}^T(I)$,

$$\mathcal{A}(\underline{u},\underline{v}) = \mathcal{B}(\underline{v}) \tag{2.17}$$

where the bilinear and linear forms are defined by:

$$\mathcal{A}(\underline{u},\underline{v}) = \int_I m(\ddot{\underline{u}},\underline{v})dt + \int_I d(\dot{\underline{u}},\underline{v})dt + \int_I k(\underline{u},\underline{v})dt + m(\dot{\underline{u}}(0^+),\underline{v}(0^+)) + k(\underline{u}(0^+),\underline{v}(0^+)) \tag{2.18}$$

$$\mathcal{B}(\underline{v}) = \int_I f(\underline{v};t)dt + m(\dot{\underline{u}}_{in},\underline{v}(0^+)) + k(\underline{u}_{in},\underline{v}(0^+)) \tag{2.19}$$

with $\underline{v}(0^+) = \lim_{t\to 0^+}\underline{v}(t)$.

The variational formulations given by (2.18) and (2.19) are the starting points for the calculation of an approximate space-time solution, which is presented in the next section.

2.2.3 Discretization of the space-time problem

2.2.3.1 Space approximation

The discretization of the spatial problem defined by the weak space formulation of equation (2.10) is performed thanks to the classical Galerkin approach, which consists in searching the solution no longer in the space $\mathcal{U}^S$ but in a finite dimensional subspace $\mathcal{U}^S_h$ of dimension n_S. This approximation space is constructed as follows:

$$\mathcal{U}^S_h(\Omega,\underline{u}^D) = \left\{ \underline{u}^h \in \mathcal{U}^S(\Omega,\underline{u}^D) \;\middle|\; \underline{u}^h(\underline{x}) = \sum_{i=1}^{n_S} \chi_i(\underline{x})\boldsymbol{u}_i(t), \quad \forall \underline{x} \in \Omega \right\} \tag{2.20}$$

where the vector space $\{\chi_1,\chi_2,....\chi_{n_S}\}$ contains the shape-functions in space associated to each nodal values of the semi-discretized displacements $[\boldsymbol{u}_1(t),\boldsymbol{u}_2(t),....\boldsymbol{u}_{n_S}(t)]$. In this thesis work, the shape functions come from the application of the *Finite Element Method* (FEM) [Zienkiewicz et al., 2000, Nikishkov, 2010, Smith et al., 2013]. The discretized space weak formulation of equation (2.10) is therefore given by:

$\forall \underline{v}^h \in \mathcal{U}^S_h(\Omega,0),\ \forall t \in I$,

$$m(\ddot{\underline{u}}^h(t),\underline{v}^h) + d(\dot{\underline{u}}^h(t),\underline{v}^h) + k(\underline{u}^h(t),\underline{v}^h) = f(\underline{v}^h;t) \tag{2.21}$$

From the above considerations, the resolution of the semi-discretized version of (2.10) consists in finding the vector of temporal nodal displacements $\underline{\boldsymbol{u}} : I \to \mathbb{R}^{n_S}$ verifying:

$$\begin{cases} \underline{\underline{\boldsymbol{M}}}\,\underline{\ddot{\boldsymbol{u}}}(t) + \underline{\underline{\boldsymbol{D}}}\,\underline{\dot{\boldsymbol{u}}}(t) + \underline{\underline{\boldsymbol{K}}}\,\underline{\boldsymbol{u}}(t) = \underline{\boldsymbol{f}}(t) \\ \underline{\boldsymbol{u}}(0) = \underline{\boldsymbol{u}}_{in} \\ \underline{\dot{\boldsymbol{u}}}(0) = \underline{\dot{\boldsymbol{u}}}_{in} \end{cases} \tag{2.22}$$

Where the mass, stiffness and damping matrices are given by $\underline{\underline{\boldsymbol{M}}}_{ij} = m(\chi_i,\chi_j)$, $\underline{\underline{\boldsymbol{K}}}_{ij} = k(\chi_i,\chi_j)$ and $\underline{\underline{\boldsymbol{D}}}_{ij} = d(\chi_i,\chi_j)$ respectively, and the vector $\underline{\boldsymbol{f}}(t) = f(\underline{\chi};t)$.

The equations developed in this section allow us to solve the problem approximately in space, but a study is still needed to approximate the time evolution of the system.

The temporal response of the semi-discretized equation (2.22) can be solved by applying incremental temporal approximation strategies. For instance, we can cite the classic Newmark scheme [Newmark, 1959] or the Runge-Kutta solver [Runge, 1895, Kutta, 1901]. These approximations do not require a global formulation of the reference problem in time as was done here for the spatial domain. However, a different temporal approximation is followed in this thesis, which is presented in the following section.

2.2.3.2 Temporal approximation

Many schemes and methods can be used for performing the integration in time. In this work, the *Time Continuous Galerkin Method* (TCGM) [Argyris and Scharpf, 1969] and the *Time Discontinuous Galerkin Method* (TDGM) [Hulbert and Hughes, 1990] are considered. However, because of its greater adaptability and robustness the TDGM is retained for the main developments of this thesis work (with the exception of chapter 5 where TCGM is used). The DGM time-integration details are given in the following lines.

Lets consider at first the time domain divided into a number N_T of open intervals of length Δt_k such that:

$$I = \bigcup_{k=1}^{N_T} \breve{I}_k \quad , \quad \text{with} \quad \breve{I}_k =]t_{k-1}, t_k[\quad ; \quad t_0 = 0 \ \text{ and } \ t_{N_T+1} = T \tag{2.23}$$

Figure 2.2 gives an illustration of this temporal division.

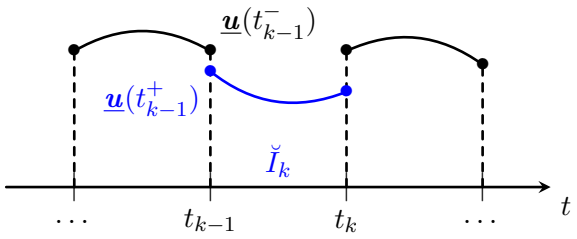


Figure 2.2: Discretization scheme in time.

Every component of the semi discretized vector $\underline{\boldsymbol{u}}(t)$ is searched in the space of discontinuous functions:

$$\mathcal{U}^{\breve{T}}(I) = \left\{ u \in \bigcup_{k=1}^{N_T} \mathcal{H}_1(\breve{I}_k) \right\} \tag{2.24}$$

where $\mathcal{H}_1(\breve{I}_k)$ designates the Sobolev space of functions u over $\breve{I}_k$ where u and ∇u are square integrable. As the name of the method suggests, the displacements, velocities and accelerations are discontinuous in the vicinity of each t_k; this gap is minimized by re-imposing the continuity in a weak sense between each time interval. With the above considerations, the bilinear $\mathcal{A}(\underline{u}, \underline{v})$ and linear $\mathcal{B}(\underline{v})$ forms of equations (2.18) and (2.19) are modified by taking into account the imposition of the continuity in a weak sense of the displacement and the velocity between the intervals $\breve{I}_{k-1}$ at t_{k-1}^- and $\breve{I}_k$ at t_{k-1}^+, $\forall k \in [1, ..., N_T]$ [Hulbert and Hughes, 1990, Boucinha et al., 2013], giving the new bilinear and linear forms $\mathcal{A}'(\underline{v}, \underline{u})$ and $\mathcal{B}'(\underline{v})$ by:

$$\mathcal{A}'(\underline{u}, \underline{v}) = \mathcal{A}(\underline{u}, \underline{v}) + \sum_{k=1}^{N_T} m(\dot{\underline{u}}(t_k^+), \dot{\underline{v}}(t_k^+)) + k(\underline{u}(t_k^+), \underline{v}(t_k^+)) \tag{2.25}$$

$$\mathcal{B}'(\underline{v}) = \mathcal{B}(\underline{v}) + \sum_{k=1}^{N_T} m(\dot{\underline{u}}(t_k^-), \dot{\underline{v}}(t_k^+)) + k(\underline{u}(t_k^-), \underline{v}(t_k^+)) \tag{2.26}$$

For the approximation of the time evolution of the problem, a finite element basis in time is introduced in this section, which will be used throughout this thesis work. Thus a finite dimensional approximation space is defined by:

$$\mathcal{U}_{\Delta t}^{T}(I) = \left\{ u \in \mathcal{U}^{\breve{T}}(I) \;\middle|\; u \in \bigcup_{k=1}^{N_T} \mathcal{H}^{p}(\breve{I}_k) \right\} \tag{2.27}$$

where $\mathcal{H}^p$ is the approximation space defined using Hermite polynomials of degree $p = 3$ and defined on each $\breve{I}_k$ in $\mathbb{R}$. These shape functions are chosen to better approximate the acceleration and therefore the dynamics behavior of the system. From a uniform element of size $\Delta t_k = t_k - t_{k-1}$, the cubic Hermite shape functions on a temporal element "k" are given by:

$\forall t \in \breve{I}_k$,

$$\begin{aligned}
\psi_1^{[k]}(t) &= \left(1 - \frac{3}{\Delta t_k^2}(t - t_{k-1})^2 + \frac{2}{\Delta t_k^3}(t - t_{k-1})^3\right) \\
\psi_2^{[k]}(t) &= \left((t - t_{k-1}) - \frac{2}{\Delta t_k}(t - t_{k-1})^2 + \frac{1}{\Delta t_k^2}(t - t_{k-1})^3\right) \\
\psi_3^{[k]}(t) &= \left(\frac{3}{\Delta t_k^2}(t - t_{k-1})^2 - \frac{2}{\Delta t_k^3}(t - t_{k-1})^3\right) \\
\psi_4^{[k]}(t) &= \left(-\frac{1}{\Delta t_k}(t - t_{k-1})^2 + \frac{1}{\Delta t_k^2}(t - t_{k-1})^3\right)
\end{aligned} \tag{2.28}$$

As a result, the approximation of a given time function $\lambda^{[k]}(t)$ on the interval $\breve{I}_k =]t_{k-1}, t_k[$ is given by:

$\forall t \in \breve{I}_k$,

$$\lambda^{[k]}(t) = \lambda_{(t_{k-1})}\psi_1^{[k]}(t) + \dot{\lambda}_{(t_{k-1})}\psi_2^{[k]}(t) + \lambda_{(t_k)}\psi_3^{[k]}(t) + \dot{\lambda}_{(t_k)}\psi_4^{[k]}(t) \tag{2.29}$$

That can be written under the following condensed form:

$$\lambda^{[k]}(t) = \underline{\psi}^{[k]}(t) \cdot \underline{\boldsymbol{\lambda}}^{[k]} \tag{2.30}$$

with the following nodal values and shape functions at element k given by:

$$\underline{\boldsymbol{\lambda}}^{[k]} = \begin{bmatrix} \lambda_{(t_{k-1})} \\ \dot{\lambda}_{(t_{k-1})} \\ \lambda_{(t_k)} \\ \dot{\lambda}_{(t_k)} \end{bmatrix} \qquad \text{and} \qquad \underline{\psi}^{[k]}(t) = \begin{bmatrix} \psi_1^{[k]}(t) \\ \psi_2^{[k]}(t) \\ \psi_3^{[k]}(t) \\ \psi_4^{[k]}(t) \end{bmatrix}$$

For the approximation of temporal functions over the whole time domain, we define the total vector of shape functions over all the temporal elements $\underline{\psi}$ as:

$$\underline{\psi}(t) = \left[\psi_1(t), \psi_2(t), \cdots, \psi_{n_T}(t)\right] \tag{2.31}$$

From the above considerations, the dimension of the space is $\dim(\mathcal{U}^{\breve{T}}) = n_T = 4N_T$. In addition, the space-time discretized displacement field is given by:

$\forall \underline{x} \in \Omega,\ \forall t \in I$,

$$\underline{u}^{\mathcal{N}}(\underline{x}, t) = \left[\underline{\chi}(\underline{x}) \otimes \underline{\psi}(t)\right] : \underline{\underline{\mathbf{u}}} \quad , \quad \underline{u}^{\mathcal{N}} \in \mathcal{U}_h^S \otimes \mathcal{U}_{\Delta t}^T \tag{2.32}$$

with $\underline{\underline{\mathbf{u}}} \in \mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T}$, a second order tuple given by:

$$\underline{\underline{\mathbf{u}}} = \left[\underline{\boldsymbol{u}}_1(t_1), \cdots, \underline{\boldsymbol{u}}_{4(k-1)+1}, \underline{\dot{\boldsymbol{u}}}_{4(k-1)+2}, \underline{\boldsymbol{u}}_{4(k-1)+3}, \underline{\dot{\boldsymbol{u}}}_{4k}, \cdots, \underline{\dot{\boldsymbol{u}}}_{n_T}(T)\right] \tag{2.33}$$

where $\otimes$ denote the Dyadic product and ":" the bi-contracted product between second order tensors:

$$\forall a, b \in \bigoplus_{d=1}^{2} \mathbb{R}^{n_d}, \quad \underline{\underline{a}} : \underline{\underline{b}} \; = \; \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} a_{i,j} \; b_{i,j} \tag{2.34}$$

From the aforementioned temporal approximation, the weak formulation of equations (2.25) and (2.26) associated with the Time Discontinuous Galerkin approach finally consists in finding $\underline{u}^{\mathcal{N}} \in \mathcal{U}_h^S \otimes \mathcal{U}_{\Delta t}^T$ such that:

$\forall \underline{v}^{\mathcal{N}} \in \mathcal{U}_h^S \otimes \mathcal{U}_{\Delta t}^T,$

$$\mathcal{A}'(\underline{u}^{\mathcal{N}}, \underline{v}^{\mathcal{N}}) = \mathcal{B}'(\underline{v}^{\mathcal{N}}) \tag{2.35}$$

To obtain the equations of the discretized system, in the following we consider the space-time approximation over a temporal element "k". Once the equations for this element are obtained, a subsequent assembly in time is applied to obtain the final discretized equations. Therefore, let us consider the semi discretized solution in time $\underline{\boldsymbol{u}}(t)$ with $t \in \breve{I}_k$ given as follows:

$$\underline{\boldsymbol{u}}(t) = \underline{\psi}^{[k]}(t) \cdot \underline{\underline{\mathbf{u}}}^{[k]} \tag{2.36}$$

Lets also consider a test function:

$$\underline{\boldsymbol{v}}(t) = \underline{\psi}^{[k]}(t) \cdot \underline{\underline{\mathbf{v}}}^{[k]} \tag{2.37}$$

where for both cases we have:

$$\begin{aligned} \underline{\underline{\mathbf{u}}}^{[k]} &= [\underline{\boldsymbol{u}}_{4(k-1)+1}, \underline{\dot{\boldsymbol{u}}}_{4(k-1)+2}, \underline{\boldsymbol{u}}_{4(k-1)+3}, \underline{\dot{\boldsymbol{u}}}_{4k}] \\ \underline{\underline{\mathbf{v}}}^{[k]} &= [\underline{\boldsymbol{v}}_{4(k-1)+1}, \underline{\dot{\boldsymbol{v}}}_{4(k-1)+2}, \underline{\boldsymbol{v}}_{4(k-1)+3}, \underline{\dot{\boldsymbol{v}}}_{4k}] \end{aligned} \tag{2.38}$$

such as:

$$\begin{aligned} \underline{\underline{\mathbf{u}}} &= \left[\underline{\boldsymbol{u}}_1(t_1) \cdots , \underbrace{\underline{\boldsymbol{u}}_{4(k-1)+1}, \underline{\dot{\boldsymbol{u}}}_{4(k-1)+2}, \underline{\boldsymbol{u}}_{4(k-1)+3}, \underline{\dot{\boldsymbol{u}}}_{4k}}_{\underline{\underline{\mathbf{u}}}^{[k]}}, \cdots , \underline{\dot{\boldsymbol{u}}}_{n_T}(T) \right] \\ \underline{\underline{\mathbf{v}}} &= \left[\underline{\boldsymbol{v}}_1(t_1) \cdots , \underbrace{\underline{\boldsymbol{v}}_{4(k-1)+1}, \underline{\dot{\boldsymbol{v}}}_{4(k-1)+2}, \underline{\boldsymbol{v}}_{4(k-1)+3}, \underline{\dot{\boldsymbol{v}}}_{4k}}_{\underline{\underline{\mathbf{v}}}^{[k]}}, \cdots , \underline{\dot{\boldsymbol{v}}}_{n_T}(T) \right] \end{aligned} \tag{2.39}$$

The details on the derivation of the discretized equations are given in appendix A. From the discretization of the bilinear and linear operators for a given temporal element k, the following equations are obtained:

$$\begin{aligned} \left[\underline{\underline{\boldsymbol{M}}} \otimes (\underline{\underline{\boldsymbol{Q}}}_k^{12} + \underline{\underline{\boldsymbol{P}}}_k^{11}) + \underline{\underline{\boldsymbol{D}}} \otimes \underline{\underline{\boldsymbol{Q}}}_k^{11} + \underline{\underline{\boldsymbol{K}}} \otimes (\underline{\underline{\boldsymbol{Q}}}_k^{10} + \underline{\underline{\boldsymbol{P}}}_k^{00}) \right] : \underline{\underline{\mathbf{u}}}^{[k]} = \\ (\underline{\underline{\boldsymbol{I}}} \otimes \underline{\underline{\boldsymbol{Q}}}_k^{10}) : \underline{\underline{\mathbf{f}}}^{[k]} + \left[\underline{\underline{\boldsymbol{M}}} \otimes \underline{\underline{\boldsymbol{R}}}_k^{11} + \underline{\underline{\boldsymbol{K}}} \otimes \underline{\underline{\boldsymbol{R}}}_k^{00} \right] : \underline{\underline{\mathbf{u}}}^{[k-1]} \end{aligned} \tag{2.40}$$

The discrete solution $\underline{\underline{\mathbf{u}}}^{[1]}$ over the initial element $\breve{I}_1$ in turn verifies:

$$\begin{aligned} \left[\underline{\underline{\boldsymbol{M}}} \otimes (\underline{\underline{\boldsymbol{Q}}}_1^{12} + \underline{\underline{\boldsymbol{P}}}_1^{11}) + \underline{\underline{\boldsymbol{D}}} \otimes \underline{\underline{\boldsymbol{Q}}}_1^{11} + \underline{\underline{\boldsymbol{K}}} \otimes (\underline{\underline{\boldsymbol{Q}}}_1^{10} + \underline{\underline{\boldsymbol{P}}}_1^{00}) \right] : \underline{\underline{\mathbf{u}}}^{[1]} = \\ (\underline{\underline{\boldsymbol{I}}} \otimes \underline{\underline{\boldsymbol{Q}}}_1^{10}) : \underline{\underline{\mathbf{f}}}^{[1]} + (\underline{\underline{\boldsymbol{M}}} . \underline{\dot{\boldsymbol{u}}}_{in}) \otimes \underline{\underline{\boldsymbol{P}}}_1^{1} + (\underline{\underline{\boldsymbol{K}}} . \underline{\boldsymbol{u}}_{in}) \otimes \underline{\underline{\boldsymbol{P}}}_1^{0} \end{aligned} \tag{2.41}$$

where the matrix related to the discretization of the temporal evolution over the interval $\breve{I}_k$ given by:

$$\underline{\underline{\boldsymbol{Q}}}_k^{ij} = \int_{\breve{I}_k} \frac{\partial^i \underline{\psi}^{[k]}(t)}{\partial t^i} \otimes \frac{\partial^j \underline{\psi}^{[k]}(t)}{\partial t^j} dt \tag{2.42}$$

and the matrices related to the weak imposition of continuity between time intervals given as follows:

$$\underline{\underline{\boldsymbol{P}}}_k^{ij} = \frac{\partial^i \underline{\psi}^{[k]}(t_{k-1})}{\partial t^i} \otimes \frac{\partial^j \underline{\psi}^{[k]}(t_{k-1})}{\partial t^j} \tag{2.43}$$

$$\underline{\underline{\boldsymbol{R}}}_k^{ij} = \frac{\partial^i \underline{\psi}^{[k]}(t_{k-1})}{\partial t^i} \otimes \frac{\partial^j \underline{\psi}^{[k-1]}(t_{k-1})}{\partial t^j} \tag{2.44}$$

$$\underline{\boldsymbol{P}}_1^1 = \underline{\psi}^{[1]}(0) \tag{2.45}$$

$$\underline{\boldsymbol{P}}_1^0 = \underline{\dot{\psi}}^{[1]}(0) \tag{2.46}$$

The full temporal evolution of the elastic problem is therefore obtained by the incremental resolution over all the temporal elements "k" of the time domain I. However, the above system of equations can also be compressed by doing a space-time assembly giving:

$$\underline{\underline{\underline{\mathbf{A}}}} : \underline{\underline{\mathbf{u}}} = \underline{\underline{\mathbf{B}}} \tag{2.47}$$

From the last equation we considered $\underline{\underline{\underline{\mathbf{A}}}}$ a fourth dimensional matrix, this is, $\underline{\underline{\underline{\mathbf{A}}}} \in \mathbb{R}^{n_S \times n_S} \otimes \mathbb{R}^{n_T \times n_T}$, additionally $\underline{\underline{\mathbf{B}}} \in \mathbb{R}^{n_S \times n_T}$. The last consideration is assumed in order to be coherent with the approximation introduced in (2.32). However in practice $\underline{\underline{\underline{\mathbf{A}}}}$ is considered to be a two dimensional matrix.

From the above compression we consider the following operations:

$$(\underline{\underline{\boldsymbol{A}}} \otimes \underline{\underline{\boldsymbol{B}}}) : \underline{\underline{\boldsymbol{C}}} = \underline{\underline{\boldsymbol{D}}} \Longleftrightarrow \sum_{j,l} A_{ij} B_{kl} C_{jl} = D_{ik} \tag{2.48}$$

Remark: Due to the time-discontinuous formulation thanks to the TDGM, the resolution of the elastic problem is performed incrementally in time for each temporal element k (one temporal element after the other). This has a great advantage, since instead of inverting an assembled matrix (which takes into account all spatio-temporal degrees of freedom) of size $n_S n_T$, only a matrix defined on a time element of size $n_S 4$, where 4 corresponds to the number of time nodes in an element k, has to be inverted. Despite this remarkable numerical advantage, it should be noted that if the spatial degrees of freedom n_S are large, even the inversion of a matrix of size $n_S 4$ can be extremely costly. In this regard, for the efficient resolution of the elastic problem in dynamics, the model reduction technique Proper Generalized Decomposition (PGD) is used together with the modal basis of the system. The PGD along with other model reduction techniques are introduced in the next chapter 3, on the other hand, the details related to the efficient resolution of the elastic problem are presented in chapter 4.

2.2.4 Nonlinear solid mechanics in dynamics

Nonlinear problems can arise from different sources, for instance geometric ones as for the case of large deformations, contacts problems or simply by considering a nonlinear constitutive relation of the material under consideration. In this thesis work the last one is only considered, this is, a nonlinearity that comes from the material behavior. This behavior is represented by the stress tensor $\underline{\underline{\sigma}}$, that under some condition is not linearly related to the displacement field through Hooke's law. In other words, we have:

$$\underline{\underline{\sigma}} = \mathcal{J}(\underline{u}) \quad \text{over} \quad \Omega \times I \tag{2.49}$$

where $\mathcal{J}(\cdot)$ is a nonlinear function.

As seen from section 2.2.1, where different approximative approaches were introduced for the resolution of linear problems (as the use of FEM in space and TDGM in time), once the spatial and temporal domain

are discretized the solution is obtained by simple determination of spatial and temporal "*nodal unknowns*". These unknowns are calculated by inverting a certain matrix, which is constructed by discretizing the equilibrium equation. The determination of the elastic solution does not pose great difficulty, since its solution is simple and straightforward. However, when the problem is nonlinear, its resolution become harder to obtain, the direct solution of this type of problem is not possible and special dedicated algorithms must be used to find the correct solution. The semi-discretized equilibrium equation for a time $t \in I$ when the problem is nonlinear due to the behavior of the material is given by:

$$\begin{cases} \underline{\underline{\boldsymbol{M}}}\,\underline{\ddot{\boldsymbol{u}}}(t) + \underline{\underline{\boldsymbol{D}}}\,\underline{\dot{\boldsymbol{u}}}(t) + \underline{\boldsymbol{g}}(t) = \underline{\boldsymbol{f}}(t) \\ \underline{\boldsymbol{u}}(0) = \underline{\boldsymbol{u}}_{in} \\ \underline{\dot{\boldsymbol{u}}}(0) = \underline{\dot{\boldsymbol{u}}}_{in} \end{cases} \tag{2.50}$$

with $\underline{\boldsymbol{g}}(t) \in \mathbb{R}^{n_S}$ calculated using the FEM method, such as it verifies:

$\forall v^h \in \mathcal{U}_h^S,\ \forall \underline{\boldsymbol{v}} \in \mathbb{R}^{n_S}$,

$$\underline{\boldsymbol{v}} \cdot \underline{\boldsymbol{g}}(t) = \int_\Omega \mathcal{J}(\underline{u}^h(t)) : \underline{\underline{\varepsilon}}(\underline{v}^h)\, d\Omega \tag{2.51}$$

The algorithms used for the solution of the nonlinear problem (2.50) are mainly based on the linearization of the equilibrium equation (2.50) in order to solve a classic linear problem iteratively, where the nonlinear solution is approximated by a sequence of linear solutions. Based in their resolution strategy, the nonlinear solvers can be classified into two groups:

- **Time-incremental algorithms**: On this group we find the family of Newton's type algorithms, which allows to solve incrementally in time the reference nonlinear problem [Simo and Hughes, 2006].
- **Non time-incremental algorithms**: On the other hand, the LArge Time INcrement method (LATIN) [Ladevèze, 1985, Ladevèze, 1996, Ladevèze, 1999], also allows the iterative solution of the nonlinear reference problem, but in this case the solution will be found over the whole space-time domain at each iteration.

More details about incremental nonlinear solvers can be found in chapter 3, where details about the non-incremental LATIN method, which is the solver considered along this thesis work, are also given.

Remark: Numerically speaking, solving nonlinear problems could be really expensive, since its computational cost increases exponentially with the number of degrees of freedom (DOF) of the discretized model, this is due to the iterative procedure needed for their resolution. Although both time-incremental and non time-incremental methods aim at solving the same nonlinear problem, this does not mean that both are equally effective in solving it. In chapter 3 (and the numerical results of chapter 4) it will be shown that the non time-incremental LATIN method presents enormous advantages over the classical incremental one, mainly due to the introduction of model order reduction techniques in its formulation that allow a huge decrease of the computational cost. However, as mentioned above, the cost of solving nonlinear problems increases with the number of DOF to be determined due to the discretization of the spatial and temporal domain. This does not exclude the LATIN method, where there are still different types of problems that remain intractable due to the high resolution times even for this optimized method, e.g. complex fatigue problems or seismic inputs of long time duration; typical problems in the domain of seismic engineering. This kind of situations are, in fact, the main problem that the present thesis intends to solve by proposing specific strategies that will be presented throughout this thesis in upcoming chapters.

2.3 Nonlinear constitutive relations considered

In this section the nonlinear constitutive relations considered in the whole manuscript are presented. The first one consists in modeling the damage generated in concrete structures (section 2.3.1) and the second one corresponds to elasto-visco-plasticity, which consists of plastic deformation, isotropic and kinematic hardening in metals (section 2.3.2). Both constitutive relations are studied to take into account the main phenomena involved regarding reinforced concrete structures (concrete plus metallic reinforcements), materials widely used for constructions, which are treated independently in this thesis. Finally, the numerical constants used for both constitutive relationships throughout the thesis are presented.

2.3.1 Isotropic damage model for concrete materials

The following sub-sections present the constitutive relations considered to model the damage on a concrete structure. In the earthquake engineering context that is considered in this work, alternate cyclic loading is expected on the solid domain. Thus, two important characteristics of concrete must be reproduced by the model namely: (i) the asymmetric traction-compression behavior referred to as *unilateral effect* [Mazars et al., 1990] and (ii) the crack-closure phenomenon. The simplified concrete modeling, developed in [Vitse et al., 2019] from previous work by [Richard and Ragueneau, 2013] and [Vassaux et al., 2015], is used here. Among other simplifications, no damage in compression is considered.

The quasi-brittle behavior of the concrete medium in tension is modeled in the classical continuum damage mechanics framework [Lemaitre and Desmorat, 2005] using a scalar damage variable denoted d. In tension, the effective stress tensor in the concrete medium classically writes (see figure 2.3a):

$$\underline{\underline{\tilde{\sigma}}}_{me} = \frac{\underline{\underline{\sigma}}_{me}}{1-d}$$

with the stress $\underline{\underline{\tilde{\sigma}}}_{me}$ following an elastic law:

$$\underline{\underline{\tilde{\sigma}}}_{me} = \mathbb{K} : \underline{\underline{\varepsilon}} \tag{2.52}$$

In compression however (see figure 2.3b), the main idea proposed in [Vassaux et al., 2015] for handling unilateral effect and progressive micro-cracks re-closure phenomena consists in the use of an additional stress tensor $\underline{\underline{\sigma}}_{cr}$. The stress tensor in the concrete representative volume element (RVE) then writes using a Kelvin-Voigt description (see figure 2.3):

$$\underline{\underline{\sigma}} = \underline{\underline{\sigma}}_{me} + \underline{\underline{\sigma}}_{cr} \tag{2.53}$$

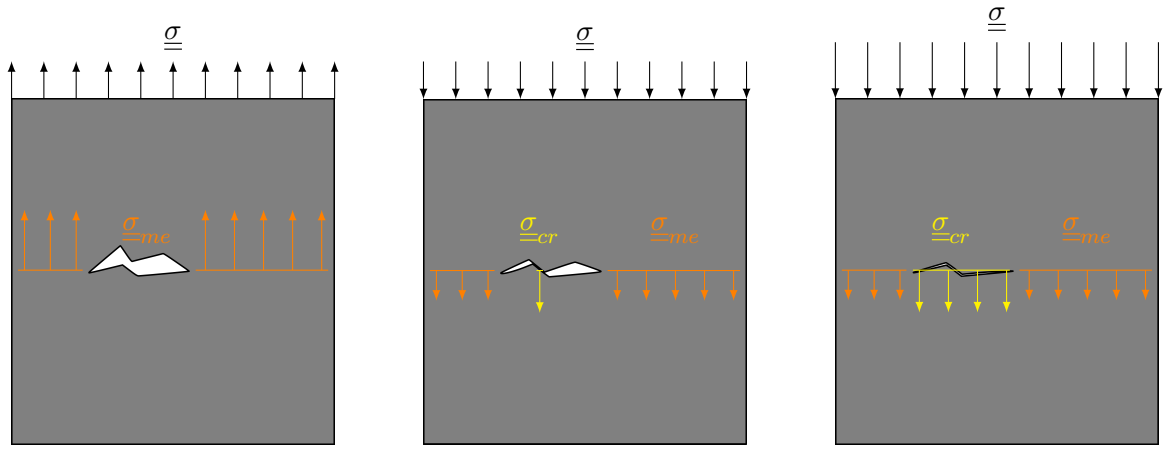
introducing $\underline{\underline{\sigma}}_{me}$ the stress in the concrete medium and $\underline{\underline{\sigma}}_{cr}$ accounting for the proportion of closed micro-cracks.

An effective stress tensor associated to gradually closed micro-cracks can then be defined as (see figure 2.3b and 2.3c.):

$$\underline{\underline{\tilde{\sigma}}}_{cr} = \frac{\underline{\underline{\sigma}}_{cr}}{d} \tag{2.54}$$

Following the Kelvin-Voigt description and using coherent indexing, the strain tensor writes:

$$\underline{\underline{\varepsilon}} = \underline{\underline{\varepsilon}}_{me} = \underline{\underline{\varepsilon}}_{cr} \tag{2.55}$$



(a) apparition of micro-cracks when loading (b) progressive cracks reclosure in compression (c) closed cracks and full stiffness recovery

Figure 2.3: Handling unilateral effect in concrete RVE – uniaxial illustration with only one micro-crack.

Section 2.3.1.1 describes how the quasi-brittle behavior in tension is modeled, assuming that $\underline{\underline{\sigma}}_{cr} = 0$. In turn section 2.3.1.2 describes how tensor $\underline{\underline{\sigma}}_{cr}$ is added to $\underline{\underline{\sigma}}_{me}$ in a progressive manner for compression loading phases. Again, no damage in compression will be taken into account in the modeling. The scalar variable d only models the micro-cracks occurring during tension loading phases.

More sophisticated modeling strategies can be taken into account to model the unilateral effect and can be used into the LATIN-PGD method without difficulties. For instance, the anisotropic damage (tensorial damage variable, handling damage in tension and compression) presented in [Souid et al., 2009] and suitable for alternative low dynamic loading can be considered.

2.3.1.1 Quasi-brittle behavior in tension

According to the continuum damage mechanics framework [Lemaitre and Chaboche, 1994], the Helmoltz free enthalpy, taken as the state potential of the cracked material, is a function of all the primal state variables and writes:

$$\Psi^{me}(d, \underline{\underline{\varepsilon}}, z) = \frac{1}{2}(1-d)\ \underline{\underline{\varepsilon}} : \mathbb{K} : \underline{\underline{\varepsilon}} + H(z) \tag{2.56}$$

introducing the classical fourth order Hooke's tensor $\mathbb{K}$ and the potential $H(z)$ and variable z associated to isotropic softening related to damage. The choice of the potential function $H(z)$ governs the post-peak behavior of the concrete medium in tension. In [Richard and Ragueneau, 2013], the following choice is made:

$$H(z) = \frac{1}{A_d}\left(-z + \log(1+z)\right) \tag{2.57}$$

the following variables are then introduced by duality writing:

$$Z = \frac{\partial \Psi^{me}}{\partial z} = \frac{\partial H(z)}{\partial z} \quad \text{and} \quad \bar{Y} = -\frac{\partial \Psi^{me}}{\partial d} = \frac{1}{2}\ \underline{\underline{\varepsilon}} : \mathbb{K} : \underline{\underline{\varepsilon}} \tag{2.58}$$

where $\bar{Y}$ is an energy release rate. Since damage mainly occurs in traction stresses, the dissipated energy released in the damage process will be considered different from $\bar{Y}$ instead we consider [Mazars, 1984]:

$$Y = \frac{1}{2}\ \langle \underline{\underline{\varepsilon}} \rangle_+ : \mathbb{K} : \langle \underline{\underline{\varepsilon}} \rangle_+ \tag{2.59}$$

where the Macaulay brackets $\langle \bullet \rangle_+$ denote the positive part of the tensor "$\bullet$" in the sense of the eigenvalues, this is:

$$\langle \underline{\underline{\varepsilon}} \rangle_+ = \sum_{i=1}^{3} \langle e_i \rangle \underline{n}_i \otimes \underline{n}_i \tag{2.60}$$

with $\langle e_i \rangle$ the positive part of the i eigenvalue associated to the eigenvector $\underline{n}_i$.

In turn Z, also homogeneous to an energy rate, is the thermodynamical force involved in the definition of the damage threshold and defines the actual size of the elastic domain. Thus, the surface threshold defined in terms of energy released through damage [Mazars, 1984] reads:

$$f(Y, Z; Y_0) = Y - (Y_0 + Z) \qquad \begin{cases} f < 0 & \text{elasticity} \\ f \geqslant 0 & \text{damage activation} \end{cases} \tag{2.61}$$

An analytical expression of damage can be obtained [Richard et al., 2010] so that the evolution laws become:

$$\bar{d} = 1 - \frac{1}{1 + A_d(Y - Y_0)} \qquad \text{and} \qquad z = -\bar{d} \tag{2.62}$$

Classical continuum damage theory based models may have difficulties to describe fracture properly; indeed, the loss of ellipticity is a recurrent issue when dealing with materials with softening behavior, leading to a localization of the deformation and mesh dependency issues [Bazant et al., 1984]. One way of overcoming such numerical difficulties is to use localization limiters [Lasry and Belytschko, 1988]. In this work the damage is regularized using a viscosity law. A strategy similar to what is described in [Allix and Deü, 1997, Allix et al., 2003, Allix, 2013] is implemented. The introduction of a delay effect leads to a new damage evolution law that can be written as:

$$\dot{d} = \frac{1}{\tau_c} \left[1 - \exp\left(-a \left[\langle \bar{d} - d \rangle_+ \right] \right) \right] \tag{2.63}$$

where $\bar{d}$ is the damage value computed using equation (2.62) and the scalars (τ_c, a) are two new parameters associated to damage regularization. The damage variable d now verifies a nonlinear first order differential equation. The maximum damage rate is given by $\dot{d}_{max} = \frac{1}{\tau_c}$ and the more or less brittle character of the damage evolution law is governed by the choice of the parameter a.

2.3.1.2 Progressive micro-crack re-closure

The unilateral effect is modelled using an auxiliary stress tensor $\underline{\underline{\sigma}}_c$ accounting for the progressive micro-cracks re-closure in compression and enabling full stiffness recovery. In this manner, even though permanently damaged in tension, the concrete medium will behave in compression almost independently from its history in tension. The progressive stiffness regain can be handled by the means of a diffeomorphism $\mathcal{F} : \mathbb{R}^3 \otimes \mathbb{R}^3 \longrightarrow \mathbb{R}^3 \otimes \mathbb{R}^3$ such that:

$$\tilde{\underline{\underline{\sigma}}}_{cr} = \mathbb{K} : \mathcal{F}(\underline{\underline{\varepsilon}}) \tag{2.64}$$

and with $\mathcal{F}(\underline{\underline{\varepsilon}})$ verifying:

$$\begin{cases} \mathcal{F}(\underline{\underline{\varepsilon}}) \sim \underline{\underline{\varepsilon}} & \text{in compression} \\ \mathcal{F}(\underline{\underline{\varepsilon}}) \sim 0 & \text{in tension} \end{cases} \tag{2.65}$$

The tangent behavior is then defined writing:

$$d\tilde{\underline{\underline{\sigma}}}_{cr} = \mathbb{K} : \left[\frac{\partial \mathcal{F}}{\partial \underline{\underline{\varepsilon}}} \right] : d\underline{\underline{\varepsilon}} \tag{2.66}$$

where $\left[\frac{\partial \mathcal{F}}{\partial \underline{\underline{\varepsilon}}}\right]$ is a fourth order tensor. In this work, following the lines of [Vassaux et al., 2015], the following function is used:

$$\underline{\underline{\tilde{\sigma}}}_{cr} = \mathbb{K} : \left[\frac{\underline{\underline{\varepsilon}} - \left(\frac{\underline{\underline{\varepsilon}}_{\max}}{a_c}\right) \log\left(1 + \exp\left(a_c \frac{\mathrm{tr}(\underline{\underline{\varepsilon}})}{\mathrm{tr}(\underline{\underline{\varepsilon}}_{\max})}\right)\right)}{\mathcal{F}(\underline{\underline{\varepsilon}})}\right] \tag{2.67}$$

where $\underline{\underline{\varepsilon}}_{\max}(t) = \max\ \mathrm{tr}(\underline{\underline{\varepsilon}}(\underline{x}, t))$ with $0 < t < T$ and a_c a constant parameter that can be modified to define a more or less progressive crack-closure ($a_c \longrightarrow \infty$ for more sudden crack-closure). This expression for $\underline{\underline{\tilde{\sigma}}}_{cr}$ verifies condition (2.65) and thus by writing:

$$\underline{\underline{\sigma}}_{cr} = d\underline{\underline{\tilde{\sigma}}}_{cr}$$

one can prove that the stiffness is fully recovered in compression. More details and uni-axial interpretations of the different parameters involved in the crack-closure relation (2.67) can be found in the reference [Vassaux et al., 2015].

2.3.1.3 Synthesis

From what has been presented in this section, the isotropic damage model considers three fundamental points, (i) the generation of damage in the structure due to tensile stresses, (ii) the consideration of unilateral effects so that the structure does not suffer damage in compression and (iii) the regularization of the damage variable. All the equations of the constitutive relations considered for isotropic damage in the concrete material are recorded in the table 2.1.

Damage calculation	Stress calculation
$Y = \frac{1}{2}\langle \underline{\underline{\varepsilon}} \rangle_+ : \mathbb{K} : \langle \underline{\underline{\varepsilon}} \rangle_+$	$\underline{\underline{\tilde{\sigma}}}_{me} = \mathbb{K} : \underline{\underline{\varepsilon}}$
$\bar{d} = 1 - \frac{1}{1+A_d(Y-Y_0)}$	$\underline{\underline{\tilde{\sigma}}}_{cr} = \mathbb{K} : \left[\underline{\underline{\varepsilon}} - \left(\frac{\underline{\underline{\varepsilon}}_{\max}}{a_c}\right) \log\left(1 + \exp\left(a_c \frac{\mathrm{tr}(\underline{\underline{\varepsilon}})}{\mathrm{tr}(\underline{\underline{\varepsilon}}_{\max})}\right)\right)\right]$
$\dot{d} = \frac{1}{\tau_c}\left[1 - \exp\left(-a\left[\langle \bar{d} - d \rangle_+\right]\right)\right]$	$\underline{\underline{\sigma}} = \underline{\underline{\sigma}}_{me} + \underline{\underline{\sigma}}_{cr} = (1-d)\underline{\underline{\tilde{\sigma}}}_{me} + d\,\underline{\underline{\tilde{\sigma}}}_{cr}$

Table 2.1: Constitutive relations for the isotropic damage in concrete material.

2.3.1.4 Constants considered in the whole manuscript

For the determination of the material parameters used for the isotropic damage model, here we consider the numerical calibration of a simple traction-compression response (at single Gauss point level) in order to reproduce a experimental data. In this sense, the reference parameters presented in table 2.2 have been chosen such as reproducing the main features of the results obtained for uni-axial tests on a double-notched concrete specimen in [Nouailletas et al., 2015]. Figure 2.4 reproduces for convenience the experimental set-up description and uni-axial response of the concrete specimen extracted from the above-mentioned reference.

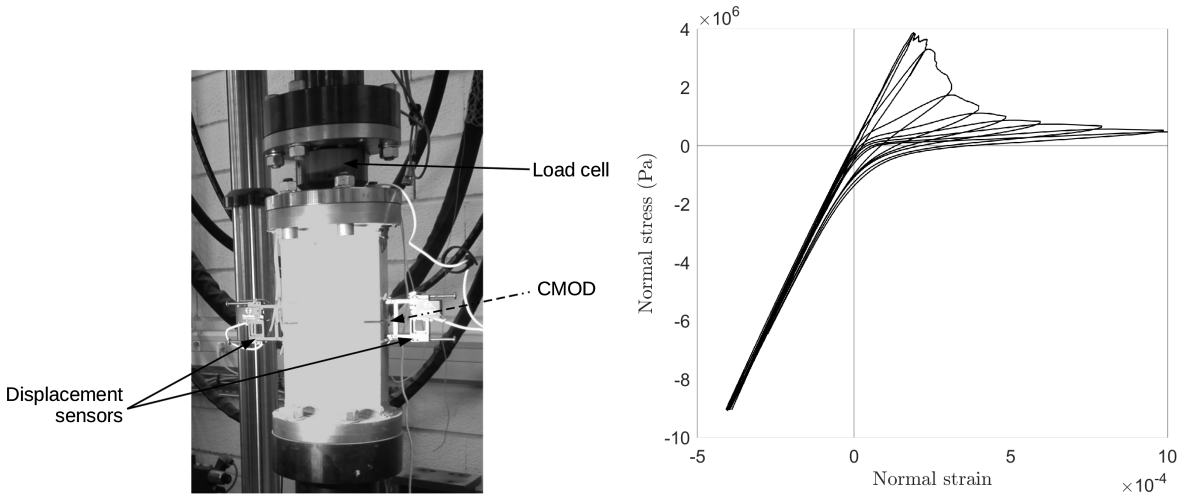


Figure 2.4: Uni-axial test results on double-notched concrete specimen described in [Nouailletas et al., 2015]. (Left) Imposed displacement in the vertical direction, two Crack Mouth Opening Displacement (CMOD) sensors are measuring the displacement between the two faces of each notch. (Right) Stress vs. strain plot in the vertical direction; the "normal strain" on abscissa is an averaged value computed from the two CMOD sensors.

Figure 2.5 shows a first illustration of the model response in a Gauss point to an uni-axial strain imposed in the x direction (with seven loops). One can particularly appreciate how the softening behavior is handled by the potential H and observe the evolution of damage taking into account (d) or not ($\bar{d}$) the damage delay of equation (2.63).

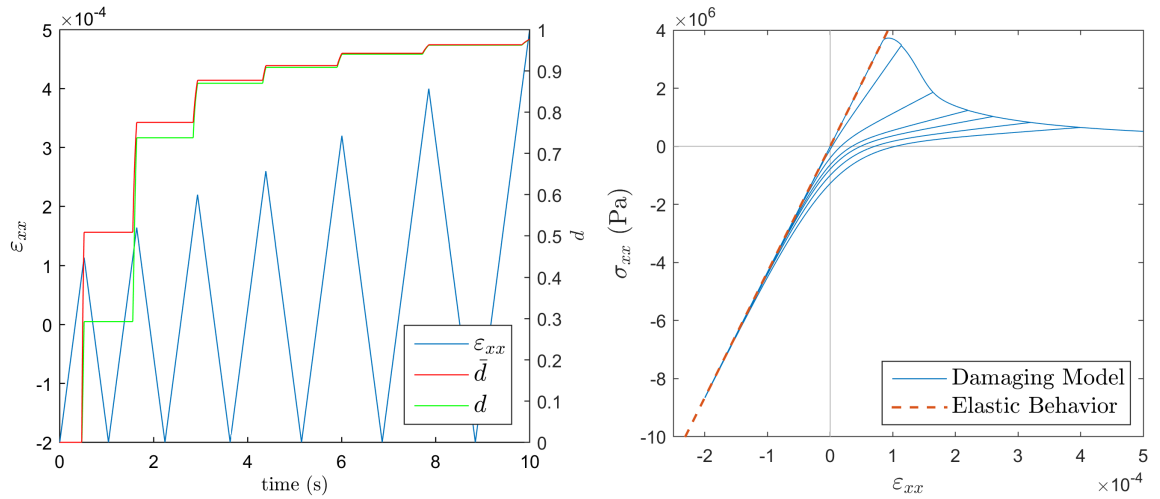


Figure 2.5: Mechanical response of the model at a Gauss point to a uni-axial tension-compression loading.

One can observe that the simplified concrete modeling considered in this work is capable of reproducing the softening behavior in tension and handling stiffness recovery in compression, and this, while taking advantage of the regularizing action of the damage delay technique. No dissipative mechanism is introduced in our simplified modeling to reproduce the hysteresis loops (unlike what can be found in [Richard and Ragueneau, 2013, Vassaux et al., 2015], that use a visco-plastic potential in their constitutive relation formulations). As previously mentioned, no damage in compression is taken into account.

Mat.	Parameters	Ref. value	Description
Concrete	ρ	$2550\ [kg/m3]$	Material density
	E	37.9 [GPa]	Young's modulus
	ν	0.2	Poisson's ratio
	Y_0	$150\ [J.m^3.kg^{-1}]$	Damage activation energy threshold
	A_d	$8.10^{-3}\ [J^{-1}.m^{-3}.kg]$	Brittleness coefficient
	τ_c	0.05 [s]	Damage delay time constant
	a	15	Damage delay exp. constant
	a_c	9	Crack-closure dimensionless constant

Table 2.2: Reference material parameters.

2.3.2 Elasto-visco-plastic constitutive relation

As mentioned previously, the behavior of steel is modeled using an elasto-visco-plastic law. In this sense, let us consider a material undergoing *kinematic* and *isotropic* hardening along with plastification. In this situation, the elastic and plastic deformation tensors $\underline{\underline{\varepsilon}}^e$ and $\underline{\underline{\varepsilon}}^p$ verify the following strain partition relation:

$$\underline{\underline{\varepsilon}} = \underline{\underline{\varepsilon}}^e + \underline{\underline{\varepsilon}}^p \tag{2.68}$$

in a Maxwell context, where $\underline{\underline{\varepsilon}}$ is the total deformation.

The constitutive relation considered and detailed below corresponds to the standard variant of the Marquis-Chaboche plasticity model ([Lemaitre and Chaboche, 1985, Marquis, 1989]) introduced in [Ladevèze, 1985, Ladevèze, 1989]. This choice of elasto-visco-plastic behavior follows the work in [Ladevèze, 1999], although another plasticity model could be considered without difficulty.

2.3.2.1 Internal variables and thermodynamical framework

The dual description (in the stress plane) of the plasticity domain involves two internal variables:

- R : a scalar variable implicated in the plasticity threshold expression and describing the actual size of the plasticity domain (*isotropic hardening*),
- $\underline{\underline{\beta}}$: a tensor variable for positioning the plasticity domain with respect to the stress state (*kinematic hardening*).

The corresponding primal variables r and $\underline{\underline{\alpha}}$ are introduced by duality and the Gibbs's free energy writes in terms of primal variables:

$$\rho\Psi \ = \ \rho\Psi^e \ + \ H(\underline{\underline{\alpha}}) \ + \ G(r) \tag{2.69}$$

involving two nonlinear potential functions $H(\underline{\underline{\alpha}})$ and $G(r)$. Primal and dual variables are linked to each other writing:

$$\underline{\underline{\beta}} \ = \ \rho\frac{\partial\Psi}{\partial\underline{\underline{\alpha}}} \quad , \quad R \ = \ \rho\frac{\partial\Psi}{\partial r} \ = \ G'(r) \tag{2.70}$$

The dissipation verifies:

$$\Phi \;=\; \underline{\underline{\sigma}} : \dot{\underline{\underline{\varepsilon}}}^{p} - R\dot{r} - \underline{\underline{\beta}} : \dot{\underline{\underline{\alpha}}} \;\geqslant\; 0 \tag{2.71}$$

In the standard version the plasticity threshold function f_s writes:

$$f_s = \sqrt{\frac{3}{2}\underline{\underline{\tau}} : \underline{\underline{\tau}}} + \frac{a}{2C}\underline{\underline{\beta}} : \underline{\underline{\beta}} - R - \sigma_y \tag{2.72}$$

with $\underline{\underline{\tau}} = \underline{\underline{\sigma}}^{D} - \underline{\underline{\beta}}$, where $\underline{\underline{\sigma}}^{D}$ correspond to the deviatoric stress tensor and σ_y is the yield stress. The elasticity domain corresponds to $f_s \leq 0$, plasticity domain is characterized by $f_s = 0$ and visco-plasticity by $f_s > 0$.

Remark: The standard version of the Marquis-Chaboche model differs from the non standard one [Lemaitre and Chaboche, 1985, Chaboche, 1989, Marquis, 1989, Lemaitre and Chaboche, 1994] in the evolution of the internal variable that describes the kinematic hardening. In this sense, an additional term $\frac{a}{2C}\underline{\underline{\beta}} : \underline{\underline{\beta}}$ in the plasticity threshold function of equation (2.72) is considered for the standard version.

2.3.2.2 Choice of the potential functions; state equations

Kinematic hardening is described using a quadratic potential writing:

$$H(\underline{\underline{\alpha}}) \;=\; \frac{1}{2}C\underline{\underline{\alpha}} : \underline{\underline{\alpha}} \tag{2.73}$$

using a constant material parameter C, that leads to:

$$\rho\frac{\partial \Psi}{\partial \underline{\underline{\alpha}}} = \underline{\underline{\beta}} = C\underline{\underline{\alpha}} \tag{2.74}$$

The isotropic potential G on the other hand, can be modeled using a exponential description:

$$G(r) \;=\; R_\infty \left[r + \frac{1}{b}\left(\exp\left(-br\right) - 1\right)\right] \tag{2.75}$$

introducing two new material constants b and R_∞. The variable R involved in the expression of the plasticity threshold then writes:

$$\rho\frac{\partial \Psi}{\partial r} = R(r) = R_\infty(1 - \exp\left(-br\right)) \tag{2.76}$$

2.3.2.3 Evolution laws

The standard formulation of the Marquis-Chaboche model is associated with the dual potential of dissipation:

$$\varphi^*(\underline{\underline{\sigma}}, \underline{\underline{\beta}}, r) = \frac{K}{n_s + 1}\left[\frac{\langle f_s(\underline{\underline{\sigma}}, \underline{\underline{\beta}}, r)\rangle_+}{K}\right]^{n_s+1} \tag{2.77}$$

with K, n_s material constants and $\langle \bullet \rangle_+$ the positive part of $\bullet$. The differentiation of the dual dissipation potential leads to the laws of evolution that writes:

$$\dot{\underline{\underline{\varepsilon}}}^{p} = \frac{\partial \varphi^*}{\partial \underline{\underline{\sigma}}} \quad , \quad -\dot{\underline{\underline{\alpha}}} = \frac{\partial \varphi^*}{\partial \underline{\underline{\beta}}} \quad , \quad -\dot{r} = \frac{\partial \varphi^*}{\partial R} \tag{2.78}$$

In this way, by considering the definition of the constant $k = K^{-n_s}$, the plastic strain rate writes:

$$\dot{\varepsilon}^p_{ij} = k\langle f_s \rangle^{n_s}_+ \left[\frac{3}{2} \frac{\tau_{ij}}{\sqrt{\frac{3}{2}\underline{\underline{\tau}} : \underline{\underline{\tau}}}} \right] \tag{2.79}$$

along with the evolution of the kinematic hardening:

$$\dot{\alpha}_{ij} = -k\langle f_s \rangle^{n_s}_+ \left[-\frac{3}{2} \frac{\tau_{ij}}{\sqrt{\frac{3}{2}\underline{\underline{\tau}} : \underline{\underline{\tau}}}} + \frac{a}{C}\beta_{ij} \right] \tag{2.80}$$

and the isotropic hardening:

$$\dot{r} = k\langle f_s \rangle^{n_s}_+ \tag{2.81}$$

2.3.2.4 Normal formulation applied to the isotropic hardening

To apply the LATIN method for the resolution of the internal variables the state equations should be described by linear operators, this is not the case of the isotropic hardening as it is shown in equation (2.76), this is:

$$R(r) = R_\infty(1 - \exp(-br))$$

In order to have a linear relation between R and r a normal formulation was introduced in [Ladevèze, 1989]. The normal formulation consists of applying the following change of variables:

$$\bar{R} = R_\infty \bar{r} = R_\infty \int_0^r \left[\frac{\left(\frac{\partial^2 G(\tilde{r})}{\partial \tilde{r}^2} \right)}{R_\infty} \right]^{1/2} d\tilde{r} \tag{2.82}$$

where:

$$\bar{r} = \int_0^r \left[\frac{\left(\frac{\partial^2 G(\tilde{r})}{\partial \tilde{r}^2} \right)}{R_\infty} \right]^{1/2} d\tilde{r} = \frac{2}{b^{\frac{1}{2}}} \left[1 - \exp(-\frac{br}{2}) \right] \tag{2.83}$$

by using (2.82) and (2.83) we can write:

$$R = R_\infty \left(\frac{\bar{R}}{R_\infty} \frac{b^{\frac{1}{2}}}{2} \right) \left(2 - \frac{\bar{R}}{R_\infty} \frac{b^{\frac{1}{2}}}{2} \right) \tag{2.84}$$

along with:

$$\dot{\bar{r}} = -k\langle f_s \rangle^{n_s}_+ \left(\frac{\bar{R}}{R_\infty} \frac{b}{2} - b^{\frac{1}{2}} \right) \tag{2.85}$$

and the plasticity threshold function finally can be written as:

$$f_s = \sqrt{\frac{3}{2}\underline{\underline{\tau}} : \underline{\underline{\tau}}} + \frac{a}{2C}\underline{\underline{\beta}} : \underline{\underline{\beta}} - R_\infty \left(\frac{\bar{R}}{R_\infty} \frac{b^{\frac{1}{2}}}{2} \right) \left(2 - \frac{\bar{R}}{R_\infty} \frac{b^{\frac{1}{2}}}{2} \right) - \sigma_y \tag{2.86}$$

Fore more information and details about the normal formulation see [Ladevèze, 1999].

2.3.2.5 Synthesis

The equations associated to elasto-visco-plastic constitutive relations are gathered in table 2.3, where the normal formulation is already taken into account for the isotropic hardening.

State laws	Evolution laws
$\underline{\underline{\sigma}} = \mathbb{K} : (\underline{\underline{\varepsilon}} - \underline{\underline{\varepsilon}}^p)$	$\dot{\underline{\underline{\varepsilon}}}^p = k\langle f_s \rangle_+^{n_s} \left[\frac{3}{2} \frac{\underline{\underline{\tau}}}{\sqrt{\frac{3}{2}\underline{\underline{\tau}} : \underline{\underline{\tau}}}} \right]$
$\underline{\underline{\beta}} = C \underline{\underline{\alpha}}$	$\dot{\underline{\underline{\alpha}}} = \dot{\underline{\underline{\varepsilon}}}^p - k\langle f_s \rangle_+^{n_s} \frac{a}{C} \underline{\underline{\beta}}$
$\bar{R} = R_\infty \bar{r}$	$\dot{\bar{r}} = -k\langle f_s \rangle_+^{n_s} \left(\frac{\bar{R}}{R_\infty} \frac{b}{2} - b^{\frac{1}{2}} \right)$

Table 2.3: Constitutive relations for elasto-visco-plastic behavior.

Those equations can be divided into *state equations* and *evolution equations*. Introducing the mapping B, and denoting $\underline{\underline{X}} = (\underline{\underline{\alpha}}, \bar{r})$ and $\underline{\underline{Z}} = (\underline{\underline{\beta}}, \bar{R})$, those relations can be written:

- *Evolution equations*:

$$\begin{bmatrix} \dot{\underline{\underline{\varepsilon}}}^p \\ \dot{\underline{\underline{X}}} \end{bmatrix} = B(\underline{\underline{\sigma}}, \underline{\underline{Z}}) \tag{2.87}$$

- *State equations*:

$$\begin{cases} \underline{\underline{\sigma}} = \mathbb{K} : (\underline{\underline{\varepsilon}} - \underline{\underline{\varepsilon}}^p) \\ \underline{\underline{Z}} = \bar{\underline{\underline{\Upsilon}}}\, \underline{\underline{X}} \end{cases} \quad , \quad \bar{\underline{\underline{\Upsilon}}} = \begin{bmatrix} C & 0 \\ 0 & R_\infty \end{bmatrix} \tag{2.88}$$

  with $\bar{\underline{\underline{\Upsilon}}}$ a linear operator.

2.3.2.6 Constants considered in the whole manuscript

Here we present the material parameters related to steel and its elasto-visco-plastic model, these constants are considered for all this thesis work. The parameters related to the steel material are presented in the table 2.4.

Mat.	Parameters	Ref. value	Description
Steel	ρ	$7850\ [kg/m3]$	Material density
	E	200 [GPa]	Young's modulus
	ν	0.3	Poisson's ratio
	σ_y	350 [MPa]	Yield stress

Table 2.4: Reference material parameters considered for steel (May vary depending on the type of steel).

In addition, the constants related to the elasto-visco-plasticity model are chosen as presented in table 2.5 [Lemaitre and Desmorat, 2005].

Parameters	Ref. value
R_∞	36 [MPa]
C	5500 [MPa]
a	250 [MPa]
b	2 [MPa]
n_s	2.5
K	1220×10^6
k	K^{-n_s}

Table 2.5: Constitutive relation parameters for elasto-visco-plasticity.

2.4 Conclusions

This chapter introduces the reference problem considered in this thesis work, which consists of a nonlinear dynamic problem in solid mechanics. First, the principal notations were introduced along with the approximation spaces by presenting the formulation of an elastic dynamic problem, where the governing discretized equations where obtained. Secondly, the nonlinear problem in dynamics was presented, where the source of nonlinearity of the problem comes from the material behavior. Finally, the main constitutive relations considered throughout this manuscript were presented, which are intended to simulate the behavior of reinforced concrete widely used in large constructions such as buildings, nuclear power plants, etc. These materials correspond to concrete and the steel frame reinforcements, whose behavior is modeled by a quasi-brittle isotropic damage in the case of concrete, which takes into account the loss of stiffness due to the existence of cracks in the material, and elasto-visco-plasticity for the steel reinforcement, which simulates the plastic deformations that are highly present in metallic materials. We recall that these behaviors are treated independently throughout this thesis, where the different ideas developed are applied to both behaviors separately, although the differences and complexities of these ideas applied to each material are highlighted.

As presented in chapter 1, the main objective of this thesis work consists in the fast solution of nonlinear problems in dynamics, in this sense the next chapter 3 introduces innovative techniques called model order reduction methods, which when combined with nonlinear solvers allow the efficient resolution of nonlinear problems. The main model order reduction methods, as well as the nonlinear solvers that use these models in their formulation, are presented in this chapter.

Chapter 3

Overview of Model Order Reduction techniques and their application for solving nonlinear problems

This chapter gives an overview of the different Model Order Reduction (MOR) strategies. Those strategies propose the use of a so-called low-rank approximation for finding an accurate and cheap solution to linear problems. Yielding methodologies for solving nonlinear evolutionary problems, possibly including parameters dependency, are also presented.

Contents

3.1 Introduction **27**
3.2 Low-rank approximation for time-dependent/parametric problems **27**
3.3 Incremental nonlinear solvers that use low-rank approximations **40**
3.4 Non incremental nonlinear solver that use low-rank approximations: the LATIN-PGD framework **44**
3.5 Conclusions **52**

3.1 Introduction

When solving large evolutionary problems, i.e. when the sizes of the spatial and temporal integration domains are big, known numerical issues can be encountered. On the one hand, there is a large memory resources demand due to all data that must be saved and, on the other hand, the inherent limitation of the processor speed can lead to very long computational times. Model order reduction techniques arise as some of the most efficient strategies for reducing the computational costs and enable memory savings when dealing with those kinds of problems. The main idea is based on the approximation of a linear solution by a low-rank decomposition taking advantage of the redundancy of the sought solution.

In the linear range, Model Order Reduction (MOR) techniques can be classified into two main groups: the *a posteriori* and *a priori* techniques. The objective of the present chapter is to give an overview of the methods corresponding to these two groups for the resolution of solid mechanics problems. In a second time, nonlinear solvers built around those MOR techniques are presented.

3.2 Low-rank approximation for time-dependent/parametric problems

In the present section, an overview of the state-of-the-art of linear MOR techniques is given. The *a posteriori* MOR Proper Orthogonal Decomposition (POD) and the *a priori* MOR that is widely used in this thesis, the Proper Generalized Decomposition (PGD), are presented. But before that, some definitions on low-rank approximation and corresponding approximation spaces for linear problems are given.

3.2.1 Approximation of rank m

The key idea is to reduce the product space $\mathcal{U}^S \otimes \mathcal{U}^T$ where the solution of a linear problem is sought (see chapter 2). Thus, for time-dependent problems, the reference solution $\underline{u}(\underline{x},t)$ is sought as:

$$\underline{u}(\underline{x},t) \approx \underline{u}_m(\underline{x},t) = \sum_{i=1}^{m} \underline{w}_i(\underline{x})\lambda_i(t) = \underline{\underline{W}}_m(\underline{x}) \cdot \underline{\Lambda}_m(t) \tag{3.1}$$

where:

$$\underline{\underline{W}}_m(\underline{x}) = \Big[\underline{w}_1(\underline{x}), \underline{w}_2(\underline{x}), ..., \underline{w}_m(\underline{x})\Big] \qquad , \qquad \underline{\Lambda}_m(t) = \Big[\lambda_1(t), \lambda_2(t), ..., \lambda_m(t)\Big]^T \tag{3.2}$$

The functions $\underline{w}_i(\underline{x}) \in \mathcal{U}^S$ are called space generalized eigenfunctions (or spatial modes) and the functions $\lambda_i(t) \in \mathcal{U}^T$ are called temporal generalized eigenfunctions (or temporal modes). The total number of functions m is called the rank of the approximation.

Now assuming that the linear reference problem is discretized using classical FEs in space and Time Discontinuous Galerkin Method (TDGM) in time, the numerical solution $\underline{u}^{\mathcal{N}}(\underline{x},t)$ belonging to the approximation space $\mathcal{U}_h^S \otimes \mathcal{U}_{\Delta t}^T$ of finite dimension writes:

$$\underline{u}^{\mathcal{N}}(\underline{x},t) = \big[\underline{\chi}(\underline{x}) \otimes \underline{\psi}(t)\big] : \underline{\underline{\mathbf{u}}} \quad , \quad \underline{\underline{\mathbf{u}}} \in \mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T} \tag{3.3}$$

introducing the vector of shape function basis $\underline{\chi}(\underline{x})$ in space and $\underline{\psi}(t)$ in time, according to the notations introduced in chapter 2 and where ":" denotes the bi-contraction operation (see chapter 2, section 2.2.3.2).

The discretized low-rank approximation can thus be written:

$$\underline{u}_m^{\mathcal{N}}(\underline{x},t) = \big[\underline{\chi}(\underline{x}) \otimes \underline{\psi}(t)\big] : \underbrace{\left(\sum_{i=1}^{m} \underline{\boldsymbol{w}}_i \otimes \underline{\boldsymbol{\lambda}}_i\right)}_{\underline{\underline{\mathbf{u}}}_m} \quad \text{with} \quad \begin{cases} \underline{\boldsymbol{w}}_i \in \mathbb{R}^{n_S} \\ \underline{\boldsymbol{\lambda}}_i \in \mathbb{R}^{n_T} \end{cases} \tag{3.4}$$

with $\otimes$ the dyadic product and where $\underline{\underline{\mathbf{u}}}_m$ denotes the discretized low-rank approximation of matrix $\underline{\underline{\mathbf{u}}}$.

The problem consists therefore in finding an approximation of $\underline{u}(\underline{x},t)$ in the subset $\mathcal{R}_m$ of space-time representations with separated variables of rank m defined as:

$$\mathcal{R}_m = \left\{ \underline{u}_m \in \mathcal{U}^S \otimes \mathcal{U}^T \;\middle|\; \underline{u}_m(\underline{x},t) = \sum_{i=1}^{m} \underline{w}_i(\underline{x})\lambda_i(t) \quad \text{with} \quad \underline{w}_i \in \mathcal{U}^S, \lambda_i \in \mathcal{U}^T \right\} \tag{3.5}$$

For the case of a solution sought in FEM approximation space, of finite dimension, associated low-rank approximation space writes:

$$\mathcal{R}_m^{\mathcal{N}} = \left\{ \underline{u}_m^{\mathcal{N}} \in \mathcal{U}_h^S \otimes \mathcal{U}_{\Delta t}^T \;\middle|\; u_m^{\mathcal{N}}(\underline{x},t) = \left[\underline{\chi}(\underline{x}) \otimes \underline{\psi}(t)\right] : \underline{\underline{\mathbf{u}}}_m \quad \text{with} \quad \underline{\underline{\mathbf{u}}}_m \in \boldsymbol{\mathcal{R}}_m \right\} \tag{3.6}$$

where $\boldsymbol{\mathcal{R}}_m$ is the subset of space-time decompositions of rank m defined by:

$$\boldsymbol{\mathcal{R}}_m = \left\{ \underline{\underline{\mathbf{u}}}_m \in \mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T} \;\middle|\; \underline{\underline{\mathbf{u}}}_m = \sum_{i=1}^{m} \underline{\boldsymbol{w}}_i \otimes \underline{\boldsymbol{\lambda}}_i \quad \text{with} \quad \underline{\boldsymbol{w}}_i \in \mathbb{R}^{n_S},\ \underline{\boldsymbol{\lambda}}_i \in \mathbb{R}^{n_T} \right\} \tag{3.7}$$

In addition, the following canonical scalar product together with associated L^2-norms are introduced and applied to functions $\underline{u}$ and $\underline{v}$ of the space $\mathcal{U}^S \otimes \mathcal{U}^T$ as follows:

$$\langle \underline{v}, \underline{u} \rangle_\Omega = \int_\Omega \underline{v} \cdot \underline{u}\, d\Omega, \qquad \|\underline{v}\|_\Omega = \sqrt{\langle \underline{v}, \underline{v} \rangle_\Omega} \tag{3.8a}$$

$$\langle \underline{v}, \underline{u} \rangle_I = \int_I \underline{v} \cdot \underline{u}\, dt, \qquad \|\underline{v}\|_I = \sqrt{\langle \underline{v}, \underline{v} \rangle_I} \tag{3.8b}$$

$$\langle\langle \underline{v}, \underline{u} \rangle\rangle = \int_\Omega \int_I \underline{v} \cdot \underline{u}\, d\Omega dt, \qquad |||\underline{v}||| = \sqrt{\langle\langle \underline{v}, \underline{v} \rangle\rangle} \tag{3.8c}$$

The low-rank approximation can be constructed in two different ways:

1. ***A posteriori* construction**: This strategy corresponds to the construction of the approximation $\underline{u}_m^{\mathcal{N}}(\underline{x},t) \in \mathcal{R}_m$ by minimizing an *a posteriori* norm $J^\nabla(\underline{u}_m^{\mathcal{N}}, \underline{u}^{\mathcal{N}})$ that measures the distance between the approximation and the reference finite element solution field $\underline{u}^{\mathcal{N}}(\underline{x},t)$ which is known.

2. ***A priori* construction**: The field $\underline{u}^{\mathcal{N}}(\underline{x},t)$ remains unknown and the approximation $\underline{u}_m^{\mathcal{N}}(\underline{x},t) \in \mathcal{R}_m$ is calculated only by using the governing equations of the reference problem (i.e the bilinear and linear forms ($\mathcal{A}$ and $\mathcal{B}$ respectively) defined for the elastic problem in chapter 2) by minimizing an *a priori* norm $J^\triangle(\underline{u}_m^{\mathcal{N}}, \mathcal{A}, \mathcal{B})$, that measures the residue of the reference problem.

The description of these two strategies is the main objective of the two following sections.

3.2.2 *A posteriori* Model Order Reduction - *Proper Orthogonal Decomposition* (POD)

The decomposition known as *Proper Orthogonal Decomposition* (POD), Karhunen-Loève decomposition (KLD) [Karhunen, 1946, Kosambi, 1943, Loeve, 1948, Obukhov, 1954] and also called *Principal Component Analysis* [Pearson, 1901] (PCA), consists in calculating a decomposition of the reference solution $\underline{u}(\underline{x},t)$ which is optimal in the sense of a given norm $|||\cdot|||_\nabla$. Many publications as [Wu et al., 2003] show the equivalence between the POD, KLD, PCA and their connexion with the *Singular Value Decomposition* (SVD) in some particular cases.

3.2.2.1 Best approximation of rank m

To illustrate the concepts, lets consider that the displacement solution $\underline{u}(\underline{x},t)$ of a given problem is already known, and we want to approximate it by a low-rank decomposition. The best approximation of rank m is therefore defined as the one that minimizes a given distance to the reference solution $\underline{u}(\underline{x},t)$ in the sense of the norm $|||\cdot|||_\triangledown$ defined on $\mathcal{U}^S \otimes \mathcal{U}^T$:

$$\underline{u}_m = \underset{\underline{u}^*(\underline{x},t)\in\mathcal{R}_m}{\text{arg min}}\ J^\triangledown(\underline{u}^*;\underline{u}) = \underset{\underline{u}^*(\underline{x},t)\in\mathcal{R}_m}{\text{arg min}}\ |||\underline{u}(\underline{x},t)-\underline{u}^*(\underline{x},t)|||_\triangledown^2 \tag{3.9}$$

where the norm $|||\cdot|||_\triangledown$ is associated to the scalar product $\langle\langle\cdot,\cdot\rangle\rangle_\triangledown$. The choice of different norms $|||\cdot|||_\triangledown$ defined over $\mathcal{U}^S\otimes\mathcal{U}^T$ for the minimization problem gives different approximations $\underline{u}_m$, optimal in the sense of the chosen norm. In what follows the canonical norm $|||\cdot|||^2 = \langle\langle\cdot,\cdot\rangle\rangle$ is chosen.

In this context, the Karhunen-Loève decomposition is obtained as follows:

$$\underline{u}_m = \underset{\substack{\underline{w}_i\in\mathcal{U}^S\\ \lambda_i\in\mathcal{U}^T}}{\text{arg min}} \left|\left|\left|\underline{u}(\underline{x},t)-\sum_{i=1}^m \underline{w}_i(\underline{x})\lambda_i(t)\right|\right|\right|^2 \tag{3.10}$$

subjected to the orthonormalization of the spatial functions:

$\forall (i,j)\in[1,...,m]$,

$$\int_\Omega \underline{w}_i\cdot\underline{w}_j\, d\Omega = \delta_{ij} \quad \text{with} \quad \delta_{ij}=\begin{cases}1 & \text{if} \quad i=j\\ 0 & \text{if} \quad i\neq j\end{cases} \tag{3.11}$$

It is possible to show that the problem (3.10) can be reduced to an eigenvalue problem as follows:

$\forall i\in[1,...,m]$,

$$G^\Omega(\underline{w}_i) = \varsigma_i\underline{w}_i \qquad \text{and} \qquad G^I(\lambda_i)=\varsigma_i\lambda_i \tag{3.12}$$

where the operators $G^\Omega(\underline{w}_i)$ and $G^I(\lambda_i)$ are called spatial and temporal correlation operators respectively; the eigenvalue ς_i is associated to the proper functions $\underline{w}_i(\underline{x})$ and $\lambda_i(t)$. Indeed, by developing the expression (3.10):

$$J=\left|\left|\left|\underline{u}-\sum_{i=1}^m\underline{w}_i\lambda_i\right|\right|\right|^2 = \langle\langle\underline{u},\underline{u}\rangle\rangle - 2\sum_{i=1}^m\langle\langle\underline{u},\underline{w}_i\lambda_i\rangle\rangle + \sum_{i=1}^m\sum_{j=1}^m\langle\langle\underline{w}_i\lambda_i,\underline{w}_j\lambda_j\rangle\rangle \tag{3.13}$$

and minimizing it with respect to the spatial and temporal functions ($\underline{w}_i$ and λ_i respectively) and considering the orthonormalization condition (3.11), one obtains:

$\forall i\in[1,...,m]$,

$$\forall\underline{x},\quad \frac{dJ}{d\underline{w}_i} = -2\langle\underline{u}(\underline{x},t),\lambda_i\rangle_I + 2\langle\lambda_i,\lambda_i\rangle_I\,\underline{w}_i = 0 \Longleftrightarrow \underline{w}_i(\underline{x}) = \frac{\langle\underline{u}(\underline{x},t),\lambda_i\rangle_I}{\langle\lambda_i,\lambda_i\rangle_I} \tag{3.14}$$

$$\forall t,\quad \frac{dJ}{d\lambda_i} = -2\langle\underline{u}(\underline{x},t),w\rangle_\Omega + 2\langle\underline{w}_i,\underline{w}_i\rangle_\Omega\,\lambda_i = 0 \Longleftrightarrow \lambda_i(t) = \frac{\langle\underline{u}(\underline{x},t),\underline{w}_i\rangle_\Omega}{\langle\underline{w}_i,\underline{w}_i\rangle_\Omega} \tag{3.15}$$

By introducing the equation (3.15) into (3.14) we finally obtain:

$$\underbrace{\langle\underline{u}(\underline{x},t),\frac{\langle\underline{u}(\underline{x},t),\underline{w}_i\rangle_\Omega}{\langle\underline{w}_i,\underline{w}_i\rangle_\Omega}\rangle_I}_{G^\Omega(\underline{w}_i)} = \underbrace{\langle\frac{\langle\underline{u}(\underline{x},t),\underline{w}_i\rangle_\Omega}{\langle\underline{w}_i,\underline{w}_i\rangle_\Omega},\frac{\langle\underline{u}(\underline{x},t),\underline{w}_i\rangle_\Omega}{\langle\underline{w}_i,\underline{w}_i\rangle_\Omega}\rangle_I}_{\varsigma_i\geq 0}\underline{w}_i \tag{3.16}$$

In a similar way, the operator $G^I(\lambda_i)$ can be obtained by introducing the equation (3.14) into (3.15).

The Karhunen-Loève decomposition verifies convergence to the original function $\underline{u}(\underline{x},t)$. Indeed, by introducing the low-rank approximation $\underline{u}_m \in \mathcal{R}_m$ in equation (3.13), and considering the orthonormalization condition (3.11), the truncation error verifies the following convergence property [Karhunen, 1946]:

$$\left|\left|\left|\underline{u}(\underline{x},t) - \sum_{i=1}^{m} \underline{w}_i(\underline{x})\lambda_i(t)\right|\right|\right|^2 = |||\underline{u}(\underline{x},t)|||^2 - \sum_{i=1}^{m} \varsigma_i \quad \underset{m\to\infty}{\longrightarrow} \quad 0 \tag{3.17}$$

with the eigenvalues verifying $\varsigma_1 \geq \varsigma_2 \geq ... \geq 0$. In practice, a low number of POD modes is required to correctly approximate a given solution, so, in general, a truncation of the POD decomposition is considered taking into account only the first m terms, with m a small value.

3.2.2.2 Discrete *a posteriori* construction of the approximation

In the present section the discrete version of the POD is shown. To illustrate the methodology we consider the data to be approximated as the discretized space-time displacement $\underline{\underline{\mathbf{u}}} \in \mathbb{R}^{n_S} \otimes \mathbb{R}^{n_T}$, which is known. Following (3.10), the discrete POD decomposition must be constructed such as:

$$\underline{\underline{\mathbf{u}}}_m = \underset{\underline{\underline{\mathbf{u}}}^* \in \boldsymbol{\mathcal{R}}_m}{\arg\min} \left|\left|\left|\underline{\underline{\mathbf{u}}}^* - \underline{\underline{\mathbf{u}}}\right|\right|\right|^2 \tag{3.18}$$

where the norm $|||\underline{\underline{\mathbf{u}}}||| = \sqrt{\underline{\underline{\mathbf{u}}} : \underline{\underline{\mathbf{u}}}}$ corresponds to the Euclidean norm.

The discretized approximation basically postulates that the n_T observed variables of matrix $\underline{\underline{\mathbf{u}}}$ can be approximately obtained from a linear transformation $\underline{\underline{\boldsymbol{W}}}$ of m unknown reduced latent variables, which in this example corresponds to the discretized POD time functions $\underline{\underline{\boldsymbol{\Lambda}}} = [\underline{\boldsymbol{\lambda}}_1, \underline{\boldsymbol{\lambda}}_2, ..., \underline{\boldsymbol{\lambda}}_m]^T \in \mathbb{R}^m \otimes \mathbb{R}^{n_T}$, such that:

$$\underline{\underline{\mathbf{u}}} \approx \underline{\underline{\boldsymbol{W}}}\,\underline{\underline{\boldsymbol{\Lambda}}} \tag{3.19}$$

with the matrix $\underline{\underline{\boldsymbol{W}}} \in \mathbb{R}^{n_S} \otimes \mathbb{R}^m$ given by:

$$\underline{\underline{\boldsymbol{W}}} = \left[\underline{\boldsymbol{w}}_1, \underline{\boldsymbol{w}}_1, \cdots, \underline{\boldsymbol{w}}_m\right] \tag{3.20}$$

In the context of the POD, the reduced basis vectors $\underline{\boldsymbol{w}}_i$, $\forall i \in [1, ..., m]$ are determined by calculating the eigenvectors associated to the highest m eigenvalues of the following correlation matrix $\underline{\underline{\boldsymbol{C}}} \in \mathbb{R}^{n_S} \times \mathbb{R}^{n_S}$ [Lee and Verleysen, 2007]:

$$\underline{\underline{\boldsymbol{C}}} = \underline{\underline{\mathbf{u}}}\,\underline{\underline{\mathbf{u}}}^T \tag{3.21}$$

where the obtained spatial terms verify the orthonormalization property, that is:

$$\underline{\underline{\boldsymbol{I}}} = \underline{\underline{\boldsymbol{W}}}^T\,\underline{\underline{\boldsymbol{W}}} \tag{3.22}$$

where $\underline{\underline{\boldsymbol{I}}}$ is a identity matrix of size $m \times m$.

Once the discretized spatial functions have been obtained, due to their orthonormalization property, the temporal terms are simply obtained as follows:

$$\underline{\underline{\boldsymbol{\Lambda}}} = \underline{\underline{\boldsymbol{W}}}^T\,\underline{\underline{\mathbf{u}}} \tag{3.23}$$

From the above developments we can therefore write:

$$\underline{\underline{\mathbf{u}}} = \underbrace{\sum_{i=1}^{m} \underline{\boldsymbol{w}}_i \otimes \underline{\boldsymbol{\lambda}}_i}_{\underline{\underline{\mathbf{u}}}_m} + \underbrace{\sum_{j=m+1}^{n_S} \underline{\boldsymbol{w}}_j \otimes \underline{\boldsymbol{\lambda}}_j}_{\approx 0} \tag{3.24}$$

where $n_S - m$ terms are left out for defining the approximation $\underline{\underline{\mathbf{u}}}_m$.

Remark: It should be noted that the correlation matrix can also be determined as $\underline{\underline{\mathbf{u}}}^T \underline{\underline{\mathbf{u}}}$, where in this case the eigenvectors of this matrix would be the temporal functions.

3.2.3 Galerkin projection onto a reduced space-basis in linear dynamics problems

One of the interesting applications of POD is in efficient problem solving. Indeed, for evolutionary problems, once the solution ($u(\underline{x}, t)$ in our case) is determined for a given parametric set, truncated POD spatial basis $(\underline{w}_i)_{i=1}^m$ can be constructed using this solution over specific "snapshots" in time (where the number of snapshots is smaller than the total discretized temporal DOFs) as shown in section 3.2.2.2. Once the spatial functions of the truncated POD decomposition are determined, the idea is to reduce the equations needed for the calculation of the reference problem (and approximate the new solution) for a given new set of parameters by using the spatial basis $(\underline{w}_i)_{i=1}^m$. In other words, the spatial approximation space of rank m is reduced to:

$$\mathcal{U}_m^S = \text{span}\{\underline{w}_i\}_{i=1}^m \tag{3.25}$$

Now using this space and considering, for instance, the weak semi-discretized formulation in space introduced in chapter 2 (equation (2.10)), we obtain the following reduced equations:

$$\sum_{j=1}^{m}\sum_{i=1}^{m}\left[m(\underline{w}_i, \underline{w}_j)\ddot{\lambda}_i(t) + d(\underline{w}_i, \underline{w}_j)\dot{\lambda}_i(t) + k(\underline{w}_i, \underline{w}_j)\lambda_i(t)\right] = \sum_{j=1}^{m} f(\underline{w}_j; t) \tag{3.26}$$

which can be converted into the following discretized equation when using the finite element method in space:

$$\underbrace{\left(\underline{\underline{\boldsymbol{W}}}^T \underline{\underline{\boldsymbol{M}}}\, \underline{\underline{\boldsymbol{W}}}\right)}_{\underline{\underline{\bar{\boldsymbol{M}}}}} \ddot{\underline{\Lambda}}_m(t) + \underbrace{\left(\underline{\underline{\boldsymbol{W}}}^T \underline{\underline{\boldsymbol{D}}}\, \underline{\underline{\boldsymbol{W}}}\right)}_{\underline{\underline{\bar{\boldsymbol{D}}}}} \dot{\underline{\Lambda}}_m(t) + \underbrace{\left(\underline{\underline{\boldsymbol{W}}}^T \underline{\underline{\boldsymbol{K}}}\, \underline{\underline{\boldsymbol{W}}}\right)}_{\underline{\underline{\bar{\boldsymbol{K}}}}} \underline{\Lambda}_m(t) = \underbrace{\underline{\underline{\boldsymbol{W}}}^T \underline{\boldsymbol{f}}(t)}_{\underline{\bar{\boldsymbol{f}}}(t)} \tag{3.27}$$

Equation (3.27) can be written in a more condensed form as:

$$\underline{\underline{\bar{\boldsymbol{M}}}}\, \ddot{\underline{\Lambda}}_m(t) + \underline{\underline{\bar{\boldsymbol{D}}}}\, \dot{\underline{\Lambda}}_m(t) + \underline{\underline{\bar{\boldsymbol{K}}}}\, \underline{\Lambda}_m(t) = \underline{\bar{\boldsymbol{f}}}(t) \tag{3.28}$$

where $\underline{\underline{\bar{\boldsymbol{M}}}}$, $\underline{\underline{\bar{\boldsymbol{D}}}}$ and $\underline{\underline{\bar{\boldsymbol{K}}}}$ correspond to the reduced mass, damping and stiffness matrices respectively, which all have a size of $m \times m$. The loading term $\underline{\bar{\boldsymbol{f}}}(t)$ of size $m \times 1$ corresponds to the reduced external excitation.

The reduced equation (3.28) obtained by using the POD method allows the reference problem to be solved inexpensively, since the size of the spatial matrices involved in the equation are drastically reduced compared to a classically discretized problem without the application of POD, i.e $m \ll n_S$.

3.2.3.1 Synthesis of the Proper Orthogonal Decomposition

After the works of [Sirovich, 1987] or [Berkooz et al., 1993], the Proper Orthogonal Decomposition has met a great success for the analysis of model order reduction applied to transient problems. Other areas where POD has been successfully applied can be mentioned such as electrodynamics [Liang et al., 2002], elastodynamics [Lülf et al., 2015], interpolation of POD basis [Lieu et al., 2005, Lieu and Farhat, 2005, Lieu and Farhat, 2007, Amsallem and Farhat, 2008], and more generally nonlinear dynamics (see for example [Kerschen et al., 2005] and the many references therein). In the nonlinear range, the transient evolution of dynamic systems can be written in the following general form:

$$\underline{\underline{\boldsymbol{M}}}\, \ddot{\underline{\boldsymbol{u}}}(t) + \underline{\boldsymbol{g}}(\underline{\boldsymbol{u}}, \dot{\underline{\boldsymbol{u}}}; t) = \underline{\boldsymbol{f}}(t) \tag{3.29}$$

where $\underline{\boldsymbol{g}}(\underline{\boldsymbol{u}}, \underline{\dot{\boldsymbol{u}}}; t)$ is a nonlinear vector function representing the internal forces. The robustness of the POD with respect to the different nonlinearities encountered has been studied in dynamics by Feeny and Kappagantu [Feeny and Kappagantu, 1998]. Let one also mention the study of error estimators for linear and nonlinear parabolic problems in [Kunisch and Volkwein, 2001, Kunisch and Volkwein, 2002].

The POD has also been used to deal with parametric problems [Christensen et al., 1999, Lieu and Lesoinne, 2004, Schmit and Glauser, 2004, Amsallem and Farhat, 2008]. In such analyses, it is assumed that the equations of the problem depend on a certain number of parameters $\theta \in I_\theta$. However, in these studies special care must be taken when choosing the parametric "snapshots", since depending on the problem studied (if it has strong variations in the boundary conditions or a strong dynamic evolution) the reduced basis could severely affect the approximation as presented in [Glüsmann and Kreuzer, 2009].

In this context, a more successful method for the definition of efficient reduced bases for the fast solution of parametric problems was developed. Referenced under the name of *Certified Reduced Basis Method* (CRBM), the selection of the "snapshots" is performed in order to certify that the generated reduced basis allows to correctly approximate the solution in the parametric domain by using error estimators. The following section 3.2.4 is dedicated to this methodology.

3.2.4 *Certified reduced basis method* (CRBM)

The resolution of parametric problems in the context of *continuum mechanics* can present some limitations. The main difficulty that arises is the computational time needed to realize parametric studies, for instance when treating large industrial problems. In this context, the *Certified Reduce Basis Method* was developed in order to decrease the computational cost by using model order reduction techniques.

The CRBM was initially presented in [Prud'Homme et al., 2002, Maday et al., 2002] and was increasingly developed since the 2000's [Prud'Homme et al., 2002, Maday and Ronquist, 2004, Grepl, 2005, Maday, 2006, Rozza et al., 2007, Nguyen, 2008, Quarteroni et al., 2011, Veys, 2014, Galvis and Kang, 2014, Chen et al., 2017, Hain et al., 2019, Abbasi et al., 2020, Nonino et al., 2021]. The method is dedicated to the fast and reliable resolution of partial differential equations, elliptic and parabolic parametrized by a vector $\theta \in I_\theta$.

The reduced basis method is based on two stages procedure, called **offline** and **online** stages. During the offline stage, the reference problem is solved using classical finite element techniques on a coarse set of parametric values. The main idea of this direct resolution is to construct a reduced basis that can approximate the solution manifold over the whole parametric domain. On the other hand, the online stage consists of a Galerkin projection of the problem's equations into the space spanned by the reduced basis calculated at the offline stage. This projection allows to drastically reduce the computational cost during the online stage and therefore allowing a fast computation on the parametric domain.

In the next subsections, the basic principles of the CRBM framework are presented considering only elliptic partial differential equations, but this strategy is also suited for solving parametric parabolic problems (see e.g. [Grepl, 2005] for more details).

3.2.4.1 General description for elliptic partial differential equations

Lets consider the case of an elliptic solid mechanic problem written in a weak form for a given time t as:

$\forall \underline{v} \in \mathcal{U}^S,\ \theta \in I_\theta,$

$$k(\underline{u}, \underline{v}; \theta) = f(\underline{v}; \theta) \tag{3.30}$$

where one seeks to compute the parametric displacement solution $\underline{u}(\underline{x}, \theta)$. Lets define the solution of

equation (3.30) as the exact solution, this solution belong to the exact solution manifold given as:

$$\mathcal{M} = \{\underline{u}(\underline{x}, \theta) \mid \forall \underline{x} \in \Omega \ , \ \ \theta \in I_\theta\} \subset \mathcal{U}^S \tag{3.31}$$

The representation of this manifold is illustrated in figure 3.1.

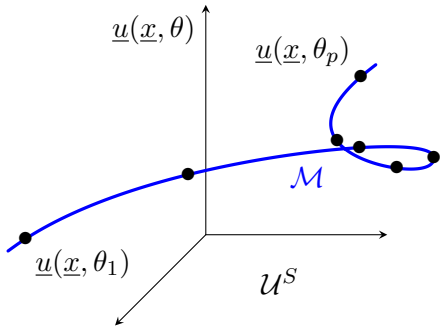


Figure 3.1: Representation of the manifold $\mathcal{M}$.

As we consider complex solid mechanics problems, finding an analytic expression of the exact solution becomes impossible, therefore we seek an approximation by using the finite element method in space (see chapter 2):

$\forall \underline{v}^h \in \mathcal{U}_h^S, \ \theta \in I_\theta,$

$$k(\underline{u}^h(\underline{x}, \theta), \underline{v}^h; \theta) = f(\underline{v}^h; \theta) \tag{3.32}$$

By consequence, we can define the approximated manifold solutions as:

$$\mathcal{M}_h = \{\underline{u}^h(\underline{x}, \theta) \mid \forall \underline{x} \in \Omega \ , \ \theta \in I_\theta\} \ \subset \ \mathcal{U}_h^S \tag{3.33}$$

As in the POD method, the idea is to build a reduced basis in an offline stage, which will then serve to effectively solve the reference problem (3.32) by means of a Galerkin projection. The big difference of the CRBM with the POD is that the reduced basis is constructed incrementally, making use of error estimators, in order to certify that the reduced basis allows to correctly approximate the solution in the online stage for all the values in the parameter domain. The presentation of the construction of the reduced base is presented in the next subsection.

3.2.4.2 Reduced basis generation

The reduced basis construction is done during the *offline* stage that consists in searching a low-rank approximation of the manifold obtained by solving the discretized reference parametric problem over some specific values of the parametric domain I_θ. These specific values are also called “snapshots”. This process is similar to the POD, however in the context of the CRBM the snapshots are selected on-the-fly based on the maximization of a reconstruction error over the whole parametric domain [Maday, 2006] as:

$$\begin{aligned} \theta_1 &= \arg\max_{\theta \in I_\theta} \left\| \underline{u}^h(\underline{x}, \theta) \right\|_{\mathcal{U}_h^S} \\ \theta_{m+1} &= \arg\max_{\theta \in I_\theta} \left\| \underline{u}^h(\underline{x}, \theta) - P_m \underline{u}^h(\underline{x}, \theta) \right\|_{\mathcal{U}_h^S} \end{aligned} \tag{3.34}$$

with the norm given by:

$\forall \underline{v}^h \in \mathcal{U}_h^S, \ \theta \in I_\theta,$

$$\left\| \underline{v}^h \right\|_{\mathcal{U}_h^S} = \sqrt{k(\underline{v}^h, \underline{v}^h; \theta)} \tag{3.35}$$

$P_m \underline{u}^h$ is the orthogonal projection into the reduced basis space constructed as:

$$\mathcal{U}_m^S = \text{span}\{\underline{u}^h(\underline{x}, \theta_1), \underline{u}^h(\underline{x}, \theta_2), ..., \underline{u}^h(\underline{x}, \theta_m)\} \tag{3.36}$$

The construction of the reduced basis in the offline step can eventually be very costly, this because the entire parameter domain must be searched to find the solutions that verify (3.34), where for each parametric value an approximation and an error must be computed. To reduce these computational times, an error estimator is introduced [Prud'Homme et al., 2002, Veroy and Patera, 2005, Rozza et al., 2007, Rozza, 2011], which is based on the hypothesis that the problem is ***compliant*** and ***coercive*** [Hesthaven, 2016], and whose determination involves the use of the *Successive Constraint Method* (SCM) which makes its determination inexpensive [Huynh et al., 2007, Huynh et al., 2010, Hesthaven, 2016]. This estimator allows to reduce the computational cost of the error evaluation in the offline and online resolution steps, allowing to speed up both stages. In addition, to further decrease the computational burden, an **affine decomposition** of the bilinear and linear forms ($k(\underline{u}, \underline{v}; \theta)$ and $f(\underline{v}; \theta)$ respectively) is used, which establish a separate variable approximation as follows:

$\forall \underline{v} \in \mathcal{U}^S,\ \theta \in I_\theta,$

$$k(\underline{u}, \underline{v}; \theta) = \sum_{i=1}^{m_k} G_i^k(\theta) k_i(\underline{u}, \underline{v}) \tag{3.37}$$

$$f(\underline{v}; \theta) = \sum_{i=1}^{m_f} G_i^f(\theta) f_i(\underline{v}) \tag{3.38}$$

This assumption is made in order to speed up the offline and online computations due to the once construction of the spatial operators involved (where the approximation of $k_i(\underline{u}, \underline{v})$ and $f_i(\underline{v})$ can be constructed independently from the parametric domain). For more complicated problems where the affine decomposition doesn't hold, the operators can be approximated by using the *Empirical Interpolation Method* (EIM) to lead to an affine decomposition approximation [Barrault et al., 2004, Grepl et al., 2007, Maday et al., 2009, Chaturantabut and Sorensen, 2009, Eftang et al., 2010, Chaturantabut and Sorensen, 2010].

The introduction of error bounds and their application for the *offline* step to assure a properly construction of the reduced basis are the key point of the CRBM. The use of error estimators allied to the use of reduced models allows to drastically decrease the CPU expenses while at the same time certifying a good quality of the approximation, making CRBM a really powerful and attractive solver for the fast resolution of parametric problems.

3.2.4.3 Synthesis of the method

The Certified Reduced Basis Method defines a general methodology for the resolution of linear parametric problems. It allows to reduce large FEM problems of dimension n_S into a low dimensional approximation by constructing iteratively an approximation space $\mathcal{U}_m^S$ based on reduced basis with guaranteed convergence properties to $\mathcal{U}_h^S$. This certified reduced space enables the approximation of the reference manifold into the whole parametric domain. The general methodology is summed up on figure 3.2 where online/offline stages are highlighted. Indeed, one of the characteristics of the CRBM approach is the partition of the computational tasks into online and offline stages. During the offline stage, the operators that don't depend on the parameter θ are calculated once and then reused along all the parametric domain by exploiting the affine decomposition of the operators. For more complex cases where the operators involved are not affine the *Empirical Interpolation Method* (EIM) can be applied to approximate the operators on a small parametric interval as an affine decomposition. The *a posteriori* error estimators are calculated in an inexpensive way by using the *Successive Constraint Method* (SCM) for the fast verification on the online stage if the approximated low-rank solution is of appropriate guaranteed quality.

We should recall that all the above works (POD and CRBM) needed to calculate the complete solution (FEM) for a given set of parameters or time steps for the construction of the reduced basis. However, in the next section 3.2.5, we present another model order reduction technique that does not require knowledge of the reference solution for the construction of the basis.

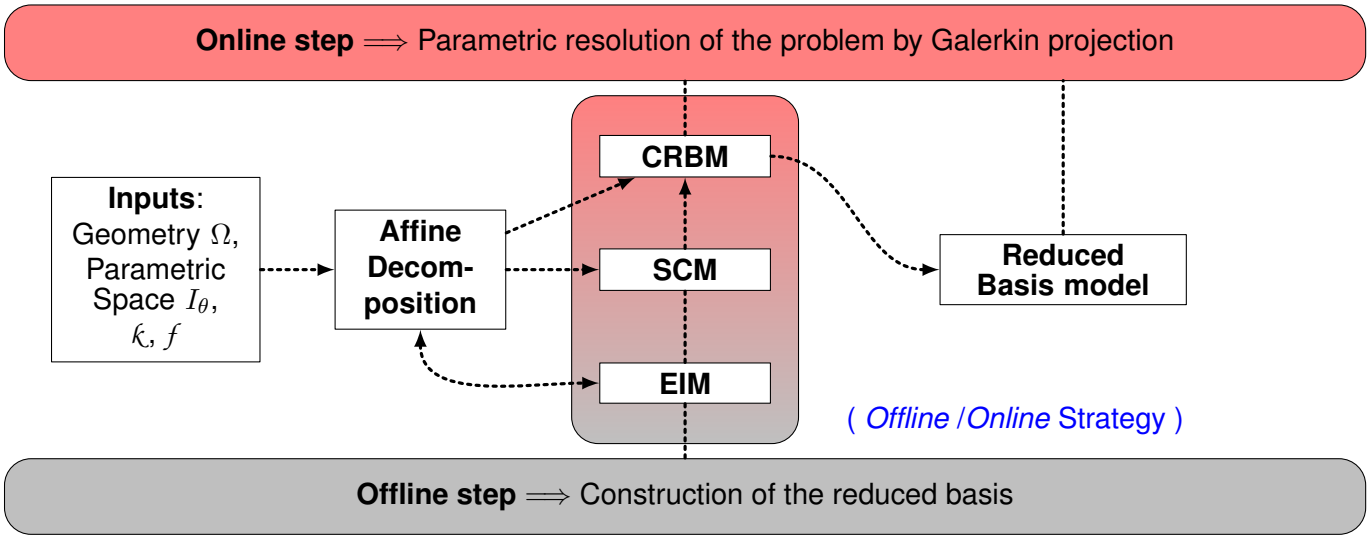


Figure 3.2: Synthesis of the Certified Reduce Basis Method.

3.2.5 *A priori* Model Order Reduction - *Proper Generalized Decomposition* (PGD)

This section is dedicated to the *Proper Generalized Decomposition* (PGD) [Chinesta et al., 2011b, Chinesta et al., 2013, Chinesta and Ladevèze, 2014], initially introduced in [Ladevèze, 1985] under the vocable *Radial Approximation* as one of the main ingredients of the LATIN framework (the LATIN methodology itself is presented later in section 3.4).

Lets recall for convenience the low-rank approximation that is sought as:

$$\underline{u}(\underline{x},t) \approx \underline{u}_m(\underline{x},t) = \sum_{i=1}^{m} \underline{w}_i(\underline{x})\lambda_i(t) = \underline{\underline{W}}_m(\underline{x}) \cdot \underline{\Lambda}_m(t)$$

Unlike the *a posteriori* techniques, the PGD strategy tends to build the decomposition $\underline{u}_m(\underline{x},t)$ (respectively $\underline{\mathbf{u}}_m$) without prior knowledge of the reference solution $\underline{u}(\underline{x},t)$; the only information considered are the operators that define the equations of the problem, naming, $\mathcal{A}$ and $\mathcal{B}$ the bilinear and linear form respectively for the case of a dynamic elastic problem ($\underline{\underline{\mathbf{A}}}$ and $\underline{\underline{\mathbf{B}}}$ respectively for the discrete case, see chapter 2). We define therefore a residual function $\mathcal{R}(\underline{u})$ from the equilibrium equation as:

$\forall \underline{v} \in \mathcal{U}^S \otimes \mathcal{U}^T$,

$$\mathcal{A}(\underline{u},\underline{v}) - \mathcal{B}(\underline{v}) = \langle\langle \underline{v}, \mathcal{A}(\underline{u}) - \mathcal{B} \rangle\rangle_\triangle = \langle\langle \underline{v}, \mathcal{R}(\underline{u}) \rangle\rangle_\triangle \tag{3.39}$$

where $\mathcal{A}$ and $\mathcal{B}$ are the Riesz representations of the bilinear $\mathcal{A}$ and linear $\mathcal{B}$ forms respectively in the Hilbert space $\mathcal{U}^S \otimes \mathcal{U}^T$ equipped with the scalar product $\langle\langle \cdot,\cdot \rangle\rangle_\triangle$. The residual function $\mathcal{R}(\underline{u})$ can be interpreted as the function that measures how much $\underline{u}$ verifies the equilibrium equation.

The functional to be minimized is therefore written using the norm $|||\cdot|||_\triangle$ over $\mathcal{U}^S \otimes \mathcal{U}^T$ and the best approximation of rank m is obtained by solving:

$$\underline{u}_m(\underline{x},t) = \underset{\underline{u}^* \in \mathcal{R}_m}{\arg\min}\, J^\triangle(\underline{u}^*;\mathcal{A},\mathcal{B}) = \underset{\underline{u}^* \in \mathcal{R}_m}{\arg\min}\, |||\mathcal{R}(\underline{u}^*)|||_\triangle^2 \tag{3.40}$$

As in the case of the *a posteriori* approximation, the choice of the norm $|||\cdot|||_{\triangle}$ gives rise to different approximations. In the following, in order to simplify the presentation, we consider the norm $|||\cdot|||$.

The problem is solved iteratively between the space and time in order to determine the functions $(\underline{w}_i(\underline{x}), \lambda_i(t))$. Many techniques have been derived for the calculation of the low-rank decomposition (see e.g. [Nouy, 2010b]). The followings subsections give a brief overview of the most standard techniques.

3.2.5.1 Rank 1 approximation: Galerkin orthogonality criterion (G)PGD

A first proposition for the calculation of the decomposition consists in imposing the minimization of the residue $\mathcal{R}(\underline{u})$ in a Galerkin orthogonality sense [Ladevèze, 1999, Ammar et al., 2006, Nouy, 2008]. Using the scalar product $\langle\langle\cdot,\cdot\rangle\rangle$ over $\mathcal{U}^S \otimes \mathcal{U}^T$ the following orthogonality condition can be expressed:

$\forall \underline{w}_1^* \in \mathcal{U}^S, \forall \lambda_1^* \in \mathcal{U}^T,$

$$\langle\langle \underline{w}_1^* \lambda_1 + \underline{w}_1 \lambda_1^*, \mathcal{R}(\underline{w}_1 \lambda_1)\rangle\rangle = 0 \tag{3.41}$$

or equivalently:

$$\mathcal{A}(\underline{w}_1 \lambda_1, \underline{w}_1^* \lambda_1 + \underline{w}_1 \lambda_1^*) = \mathcal{B}(\underline{w}_1^* \lambda_1 + \underline{w}_1 \lambda_1^*) \tag{3.42}$$

The above expression can be separated in two equations which can be solved iteratively using a fixed-point strategy, writing:

$\forall \underline{w}_1^* \in \mathcal{U}^S, \forall \lambda_1^* \in \mathcal{U}^T,$

$$\mathcal{A}(\underline{w}_1 \lambda_1, \underline{w}_1^* \lambda_1) = \mathcal{B}(\underline{w}_1^* \lambda_1) \tag{3.43}$$
$$\mathcal{A}(\underline{w}_1 \lambda_1, \underline{w}_1 \lambda_1^*) = \mathcal{B}(\underline{w}_1 \lambda_1^*) \tag{3.44}$$

The point-fixed technique consists of solving the above equations alternately, where (3.43) and (3.44) are solved assuming known temporal and spatial functions respectively, whose results come from the solution of these equations in a previous iteration of the fixed-point strategy.

3.2.5.2 Rank 1 approximation: Minimization of the residual function (R)PGD

Another strategy for the calculation of the low-rank approximation consists in the direct minimization of the residual function $\mathcal{R}(\underline{u}^*)$ [Nouy and Ladevèze, 2004, Beylkin and Mohlenkamp, 2005, Ammar et al., 2010]. This construction is the most robust due to the proof of monotonic convergence of the decomposition to the solution $\underline{u}(\underline{x}, t)$. The problem is written as follows:

$$\{\underline{w}_1, \lambda_1\} = \underset{\substack{\underline{w}_1 \in \mathcal{U}^S \\ \lambda_1 \in \mathcal{U}^T}}{\arg\min} \; |||\mathcal{A}(\underline{w}_1 \lambda_1) - \mathcal{B}|||^2 \tag{3.45}$$

which expression can be developed into:

$\forall \underline{w}_1^* \in \mathcal{U}^S, \forall \lambda_1^* \in \mathcal{U}^T,$

$$\int_{\Omega \times I} \mathcal{A}(\underline{w}_1^* \lambda_1 + \underline{w}_1 \lambda_1^*)[(\mathcal{A}(\underline{w}_1 \lambda_1) - \mathcal{B}] d\Omega dt = 0 \tag{3.46}$$

As for the case of Galerkin orthogonality PGD the above expression can be separated into a spatial and temporal problem, which can be solved iteratively.

3.2.5.3 Improvements to the Galerkin (G)PGD and the residual minimization (R)PGD strategies

Many developments were done in order to improve the quality of the PGD decomposition. The POD modes are optimal in the sense that they minimize a truncation residual (3.9) written using *a posteriori* norm $|||\cdot|||_\triangledown$. The PGD basis in turn is optimal in the sense of an *a priori* norm $|||\cdot|||_\triangle$ that is not necessarily the norm of interest. Indeed, PGD basis is computed such as minimizing equilibrium residual $\mathcal{R}(\underline{u}^*)$ in the sense of norm $|||\cdot|||_\triangle$ and in some applications, a large number of modes might be required for a good approximation of the reference solution. Most improvements intend to make the computation of the PGD basis (computed by definition using *a priori* norm) optimal in the sense of the *a posteriori* norm of interest.

1. **Minimax PGD**:

   A first attempt for minimizing the *a posteriori* functional $J^\triangledown(\underline{u}_m^{\mathcal{N}}, \underline{u}^{\mathcal{N}})$ while constructing the reduced order basis *a priori* by minimizing $J^\triangle(\underline{u}_m^{\mathcal{N}}, \mathcal{A}, \mathcal{B})$ was proposed in [Nouy, 2010a] under the name of Minimax PGD. This new strategy proposed an improvement of the convergence properties related to the Galerkin PGD (G)PGD framework by introducing an additional orthogonality criterion step of Petrov-Galerkin type. The general idea is to impose for the residual a new orthogonality condition to an additional set of modes. The optimality of the basis of interest in the sense of $|||\cdot|||_\triangledown$ can then be imposed using this second set of modes.

2. **Residual minimization in an ideal norm**:

   An improvement to the construction of the low-rank approximation by minimizing the residual function is described in [Billaud-Friess et al., 2014] and presented in the context of a parametric study. The authors propose to choose a different norm called 'ideal' for the minimization of the functional $J^\triangle(\underline{u}_m^{\mathcal{N}}, \mathcal{A}, \mathcal{B})$; more precisely, the idea is to modify $|||\cdot|||_\triangle$ for enabling a conjoint minimization of $|||\underline{u}^* - \underline{u}|||_\triangledown$ in the sense of the canonic norm of interest. This strategy was implemented in dynamics in [Boucinha et al., 2014].

3.2.5.4 Different strategies for the construction of the decomposition

As presented previously, the main problem to be solved to construct the PGD decomposition in its general form consists in minimizing *a priori* functional $J^\triangle$ such as:

$$\underline{u}_m(\underline{x}, t) = \underset{\underline{u}^* \in \mathcal{R}_m}{\arg\min}\ J^\triangle(\underline{u}^*; \mathcal{A}, \mathcal{B}) = \underset{\underline{u}^* \in \mathcal{R}_m}{\arg\min}\ |||\mathcal{R}(\underline{u}^*)|||_\triangle^2$$

with $\mathcal{R}(\underline{u})$ the residual function verifying:

$\forall \underline{v} \in \mathcal{U}^S \otimes \mathcal{U}^T$,

$$\mathcal{A}(\underline{u}, \underline{v}) - \mathcal{B}(\underline{v}) = \langle\langle \underline{v}, \mathcal{R}(\underline{u}) \rangle\rangle_\triangle$$

The determination of the PGD modes can be performed in different ways. The main strategies for the construction of this low-rank approximation are described below.

1. **Greedy construction**:

   The easiest way of constructing the low-rank approximation that minimizes equation (3.40) is based on an incremental construction, this is, mode after mode. Lets suppose that $m-1$ modes of the decomposition have already been computed and that one seeks a new mode enrichment $w_m\lambda_m$ that minimizes the actual residual:

$$\underline{w}_m\lambda_m = \underset{\substack{\underline{w}_m \in \mathcal{U}^S \\ \lambda_m \in \mathcal{U}^T}}{\arg\min}\ J^\triangle(\underline{u}_{m-1} + \underline{w}_m\lambda_m; \mathcal{A}, \mathcal{B}) \tag{3.47}$$

Once the new mode is calculated, we actualize the total approximation as $\underline{u}_m = \underline{u}_{m-1} + \underline{w}_m \lambda_m$ until we decrease the functional $J^{\triangle}(\underline{u}_m^{\mathcal{N}}, \mathcal{A}, \mathcal{B})$ to a given threshold.

2. **Direct construction**:

   Another variant for the calculation of the PGD modes consists in calculating the decomposition up to rank m directly by imposing the orthogonality of the residue with respect to span$\{\underline{w}\}_{i=1}^{m}$ and span$\{\lambda\}_{i=1}^{m}$ simultaneously, such that a complete group of spatial and temporal modes are calculated at each iteration. The *a priori* functional is minimized in consequence as:

$$\underline{\underline{W}}_m(\underline{x}) \cdot \underline{\Lambda}_m(t) = \underset{\underline{u}^* \in \mathcal{R}_m}{\text{arg min}}\ J^{\triangle}(\underline{u}^*; \mathcal{A}, \mathcal{B}) \tag{3.48}$$

3. **Greedy with actualization of modes**:

   Finally, a strategy that can be seen as a mixture between the greedy and the direct construction is presented. This last idea basically consists in calculating a new couple of modes $\underline{w}_m \lambda_m$ in a greedy way:

$$\underline{w}_m \lambda_m = \underset{\substack{\underline{w}_m \in \mathcal{U}^S \\ \lambda_m \in \mathcal{U}^T}}{\text{arg min}}\ J^{\triangle}(\underline{u}_{m-1} + \underline{w}_m \lambda_m; \mathcal{A}, \mathcal{B})$$

   Once this new mode is calculated, the groups of modes span$\{\underline{w}_i\}_{i=1}^{m}$ and span$\{\lambda_i\}_{i=1}^{m}$ are actualized [Nouy, 2010a]. In [Ammar et al., 2010] the authors advise to orthonormalize the groups of modes. On the one hand, this will avoid that the norm of the spatial mode tends towards zero while its temporal counterpart tends towards infinity or vice versa and on the other hand the orthogonalization allows to find a rich decomposition due to the fact that the new mode that will be added will be orthogonal to the previous ones.

   Finally, the whole time-functions are subjected to a final updating pass, i.e. sought such as minimizing the *a priori* functional and considering that the space functions $\underline{\underline{W}}_m(\underline{x})$ are known and fixed:

$$\underline{\Lambda}_m(t) = \underset{(\lambda_i)_{i=1}^{m} \in \mathcal{U}^T}{\text{arg min}}\ J^{\triangle}(\underline{\underline{W}}_m(\underline{x}) \cdot \underline{\Lambda}_m(t); \mathcal{A}, \mathcal{B}) \tag{3.49}$$

3.2.5.5 Synthesis of the Proper Generalized Decomposition

During the last few years, the Proper Generalized Decomposition (PGD) has gained great interest for solving partial differential equations due to its robustness, effectiveness and memory savings allowed by the nature of low-rank approximation of the reference solution. The method has been applied to a wide variety of problems covering a wide range of scientific domains; among many others, we can find:

1. **Dynamic problems**:

   In the domain of low frequency dynamics (see figure 3.3) PGD has found a successful application. For example we can find its application to the solution of elastodynamics problems [Boucinha et al., 2014], also the solution of dynamic problems using a frequency and stochastic approach [Chevreuil and Nouy, 2012].

   On the other hand, PGD has also been introduced to deal with medium-frequency dynamic problems (see figure 3.3). In this context, the PGD is applied together with *Virtual Theory of Complex Rays* (VTCR) to simulate the frequency response of the pressure field of an acoustic cavity a range of

frequencies [Barbarulo, 2012, Barbarulo et al., 2013, Barbarulo et al., 2014a, Barbarulo et al., 2014c, Cettour-Janet, 2019].

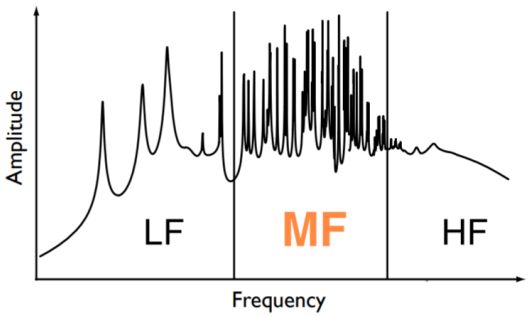


Figure 3.3: Qualitative frequency response of a system, where LF, MF and HF correspond to low-frequency, mid-frequency and high-frequency respectively.

2. **Parametric problems**:

   In addition, the PGD is very attractive when dealing with parametric problems. A natural multidimensional extension can indeed be obtained by simply adding new coordinates to the PGD representation writing:

$$\underline{u}_{m+1}(\underline{x}, t, \theta_1, ..., \theta_{n_P}) = \sum_{i=1}^{m+1} \underline{w}_i(\underline{x}) \lambda_i(t) \prod_{j=1}^{n_P} \pi_i^j(\theta_j) \tag{3.50}$$

   where the parametric functions $\pi_i^j(\theta_j)$ with θ_j the parameter variable, need to be solved iteratively in a fixed point strategy. Two principal methodologies exist, the parametric approach where each parametric function is calculated by using finite element shape functions [Chinesta et al., 2010, Chinesta et al., 2011a, Ammar et al., 2012], and the stochastic parametric approach presented in [Nouy, 2008, Chevreuil and Nouy, 2012].

   Recently new developments were done in order to consider high number of parameters in the PGD formulation and, at the same time, overcoming the so called “curse of dimensionality” (which is the exponential grow of degrees of freedom when more parameters are added) that classical PGD framework can not overcome. Before these works, the maximum value of parameters to be considered in a PGD decomposition with acceptable convergence properties was about 15 to 20. On the work presented in [Paillet et al., 2017, Paillet et al., 2018b], the number of parameters considered on the PGD decomposition raised until 1000 by using a new formulation called “parameter-multiscale PGD”, which is based on Saint-Venant’s principle.

3. **Nonlinear problems**:

   The PGD has also been used in the nonlinear range within the framework of the LArge Time INcrement (LATIN) method methodology, where the decomposition is a fundamental part of the efficiency of the solver (see e.g. [Ladevèze, 1985, Ladevèze et al., 2010b, Relun et al., 2011, Relun et al., 2013]). Indeed, parametric studies on nonlinear problems have also been carried out using the LATIN method. The great numerical efficiency is brought by the use and re-use of PGD spatial basis for the fast resolution of parametric problems as is the case of buckling problems [Boucard and Ladevèze, 1999], diffusion problems [Heyberger et al., 2012, Heyberger et al., 2013], and elasto-visco-plasticity [Néron et al., 2015].

   Additionally, the parametric study for the nonlinear case can be achieved by exploiting the decomposition (3.50), this is, by considering the full parametric solution to be approximated by a sum of

functions of separate variables, as presented in [Vitse, 2016, Vitse et al., 2019] for the case of reinforced concrete and in [Ladevèze et al., 2018] for elasto-visco-plasticity and considering a large number of parameters by using the development of [Paillet et al., 2017, Paillet et al., 2018b].

In addition, the PGD has been applied in different domains such as hyperelastic materials [Niroomandi et al., 2010], advection-diffusion-transient problems [Nouy, 2010a], high resolution digital image correlation [Passieux and Périé, 2012], inverse problems [Signorini et al., 2017], heat propagation [Zlotnik et al., 2015, Favoretto et al., 2019], domain decomposition [Huerta et al., 2018], among many others

3.3 Incremental nonlinear solvers that use low-rank approximations

The present section introduces the general framework for the resolution of nonlinear solid mechanics problems in a temporal incremental way. Once this framework presented, the principal time-incremental methods that used model reduction techniques into its formulation are presented, which consists on a variation of the Certified Reduced Basis Method applied to nonlinear problems and the A Priori Hyper-Reduction (APHR) method.

3.3.1 Newton-Raphson method: framework for the incremental resolution of nonlinear solid mechanics problems

Lets consider a classic incremental strategy for which a solution $s_k = \{\underline{u}_k, \underline{\underline{\varepsilon}}_k, \underline{\underline{\sigma}}_k\}$ is computed at time step t_k and where the solution $s_{k+1} = \{\underline{u}_{k+1}, \underline{\underline{\varepsilon}}_{k+1}, \underline{\underline{\sigma}}_{k+1}\}$ at next time step t_{k+1} is sought, given the loading terms $\underline{f}_{k+1} \in \Omega$, $\underline{f}^N_{k+1} \in \partial_N\Omega$ and $\underline{u}^D_{k+1} \in \partial_D\Omega$. The resolution of the problem mainly consists in two steps, that are:

1. The local integration of the nonlinear constitutive relations that gives $\underline{\underline{\sigma}}_{k+1}$ as function of $\underline{u}_{k+1}$ and s_k.
2. The linearization of the global equilibrium equations of the system and their resolution in an iterative way.

The global equilibrium of the structure in space at current time step t_{k+1} can be written as:
$\forall \underline{v} \in \mathcal{U}^S(\Omega, 0)$,

$$\int_\Omega \rho \ddot{\underline{u}}_{k+1} \cdot \underline{v} \, d\Omega + \int_\Omega \underline{\underline{\sigma}}_{k+1} : \underline{\underline{\varepsilon}}(\underline{v}) \, d\Omega = \int_\Omega \underline{f}_{k+1} \cdot \underline{v} \, d\Omega + \int_{\partial_N\Omega} \underline{f}^N_{k+1} \cdot \underline{v} \, dS \tag{3.51}$$

where the stress tensor $\underline{\underline{\sigma}}_{k+1}$ writes as a function of the solution s_k at previous time-step and the displacement increment $\Delta\underline{u}_k = \underline{u}_{k+1} - \underline{u}_k$ writing:

$$\underline{\underline{\sigma}}_{k+1} = \mathcal{J}(\underline{\underline{\varepsilon}}(\Delta\underline{u}_k); s_k) \tag{3.52}$$

3.3.1.1 Incremental resolution

For each time step t_{k+1}, the nonlinear problem consists in finding a kinematically admissible correction $\Delta\underline{u}_k$ (verifying the Dirichlet condition $\Delta\underline{u}^D_k$) verifying equilibrium (3.51) and the constitutive relations (3.52). Hence, the resolution of the global equilibrium consists in finding an incremental $\Delta\underline{u}_k \in \mathcal{U}^S(\Omega, \Delta\underline{u}^D_k)$ such as:

$\forall \underline{v} \in \mathcal{U}^S(\Omega, 0)$,

$$\mathcal{R}(\Delta \underline{u}_k; \underline{v}, s_k) = 0 \tag{3.53}$$

where the dynamic residual equilibrium writes:

$\forall \underline{v} \in \mathcal{U}^S(\Omega, 0)$,

$$\mathcal{R}(\Delta \underline{u}_k; \underline{v}, s_k) = \int_\Omega \rho \ddot{\underline{u}}_{k+1} \cdot \underline{v} d\Omega + \int_\Omega \mathcal{J}(\underline{\underline{\varepsilon}}(\Delta \underline{u}_k; s_k) : \underline{\underline{\varepsilon}}(\underline{v}) d\Omega - \int_\Omega \underline{f}_{k+1} \cdot \underline{v} \, d\Omega - \int_{\partial_N \Omega} \underline{f}^N_{k+1} \cdot \underline{v} \, dS \tag{3.54}$$

The residue of equation (3.54) is minimized by approaching the incremental displacement $\Delta \underline{u}_k$ as the sum of $n+1$ incremental corrections:

$$\Delta \underline{u}_k \approx \Delta \underline{u}_k^{(n+1)} = \Delta \underline{u}_k^{(n)} + \delta \underline{u}_k^{(n)} \tag{3.55}$$

where each incremental correction $\delta \underline{u}_k^{(n)}$ is solution of the equation (3.54) linearized around $\Delta \underline{u}_k^{(n)}$ in a Newton-Raphson manner (as illustrated in figure 3.4), which is written:

$\forall \underline{v} \in \mathcal{U}^S(\Omega, 0)$,

$$\mathcal{R}'(\Delta \underline{u}_k^{(n)}; \underline{v}, s_k) \cdot \delta \underline{u}_k^{(n)} = -\mathcal{R}(\Delta \underline{u}_k^{(n)}; \underline{v}, s_k) \tag{3.56}$$

The functional $\mathcal{R}' = \frac{\partial \mathcal{R}}{\partial \Delta \underline{u}_k^{(n)}}$ corresponds to the tangent linear application of the residue $\mathcal{R}$ evaluated at $\Delta \underline{u}_k^{(n)}$:

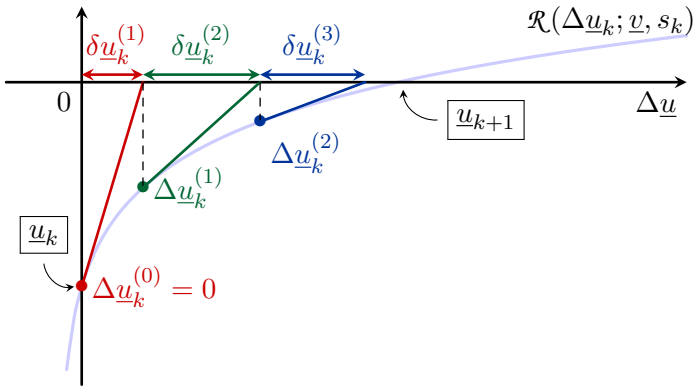


Figure 3.4: Calculation of $\Delta \underline{u}_k$ as a sum of incremental corrections $\delta \underline{u}_k^n$.

Once the problem is solved at time step t_{k+1}, the iterative resolution process is repeated for all subsequent time steps. The resolution of the linearized problem (3.56) requires the discretization of the spatial domain and the approximation of the temporal evolution. However, the level of details of its implementation is beyond the scope of this chapter, which is intended to provide a general framework for the incremental nonlinear resolution; nevertheless, the complete presentation of the incremental resolution strategy is presented in appendix B.

3.3.2 CRBM framework

As seen in section 3.2.4, the Certified Reduced Basis Method is well established for the treatment of linear parameterized partial differential equations, but they can also be applied to solve nonlinear problems [Grepl et al., 2007, Canuto et al., 2009, Jung et al., 2009]. The CRBM has largely been applied to solve nonlinear fluid mechanics problems [Peterson, 1989, Veroy et al., 2003a, Veroy et al., 2003b, Quarteroni and Rozza, 2007, Deparis and Rozza, 2009] as well as heat transfer [Rozza et al., 2009].

In what follows we will present an overview on how to apply the CRBM to a quasi-static nonlinear problem. In fact, none of the concepts presented in section 3.2.4 changes, since the classic incremental resolution of nonlinear problems requires the linearization of the governing equations, as is the case of classic Newton types methods, where all the CRBM strategy can be applied to this linearized problem.

The general CRBM framework for solving parameterized nonlinear problems will not be detailed any further. The interested reader is referred to the above mentioned citations. The following lines will only give some hints on the manner to extend the CRBM framework from elliptical problems to problems where nonlinearity is introduced. Lets thus consider that a quasi-static nonlinear problem is parameterized by a parameter $\theta \in I_\theta$. As presented in section 3.3.1, the linearization of the nonlinear problem is given by applying Newton-Raphson method (equation (3.56) by neglecting the dynamic term):

$\forall \underline{v} \in \mathcal{U}^S(\Omega, 0),\ \theta \in I_\theta,$

$$\mathcal{R}'(\Delta \underline{u}_k^{(n)}(\theta); \underline{v}, s_k(\theta), \theta) \cdot \delta \underline{u}_k^{(n)}(\theta) = -\mathcal{R}(\Delta \underline{u}_k^{(n)}(\theta); \underline{v}, s_k(\theta), \theta) \tag{3.57}$$

where we can associate the bilinear and linear forms as follows:

$\forall \underline{v} \in \mathcal{U}^S(\Omega, 0),\ \theta \in I_\theta,$

$$\mathcal{k}(\delta \underline{u}_k^{(n)}(\theta), \underline{v}; \theta) = \mathcal{R}'(\Delta \underline{u}_k^{(n)}(\theta); \underline{v}, s_k(\theta), \theta) \cdot \delta \underline{u}_k^{(n)}(\theta) \tag{3.58}$$

$$f(\underline{v}; \theta) = \mathcal{R}(\Delta \underline{u}_k^{(n)}(\theta); \underline{v}, s_k(\theta), \theta) \tag{3.59}$$

where all the formalism presented in 3.2.4 to describe the resolution of parameterized elliptical problems can be applied. The CRBM and EIM methodologies for the decomposition of operators can be used, provided that $\mathcal{l}$ is linear and $\mathcal{k}$ bilinear, continuous and coercive.

Therefore the reduction of the nonlinear problem is done by using a reduced basis $\left(\underline{w}_i^{(n)}\right)_{i=1}^{m_n}$ that depends on each iteration of the Newton-Raphson algorithm (n), and the problem is efficiently solved by projecting these basis onto the linearized equation for all the time steps.

3.3.3 *A Priori Hyper-Reduction* (APHR) method

The APHR method, firstly introduced in [Ryckelynck, 2005] (summarized in [Ryckelynck et al., 2006b]) also exploits reduced basis for approaching solutions of nonlinear problems, and since 2005, many extensions and applications have been presented, for instance for the treatment of nonlinear problems involving internal variables [Ryckelynck, 2009], damage problems [Ryckelynck et al., 2011] and parametric problems [Ryckelynck et al., 2012] among many others.

The main idea of the method consists in using a *Reduced Order Model* (ROM) for approximating the solution of a nonlinear problem. The ROM and the state evolution variables are simultaneously improved by the algorithm, thanks to an adaptive strategy. The denotation of ***hyper-reduction*** comes from the fact that a *Reduced Integration Domain* (RID) is considered for the calculation of the reduced state variables of the ROM, that in the case of space-time problems correspond to the temporal functions of the POD decomposition.

To illustrate the method, lets consider a nonlinear problem in quasi-statics, where its equilibrium equation writes:

$\forall \underline{v} \in \mathcal{U}^S(\Omega, 0),$

$$\int_\Omega \underline{\underline{\sigma}}_{k+1} : \underline{\underline{\varepsilon}}(\underline{v})\ d\Omega = \int_\Omega \underline{f}_{k+1} \cdot \underline{v}\ d\Omega + \int_{\partial_N \Omega} \underline{f}_{k+1}^N \cdot \underline{v}\ dS \tag{3.60}$$

where a nonlinear constitutive relation is considered for the stress tensor as presented in equation (3.52). By discretizing equation (3.60) using the finite element method in space, and by defining a residual equilibrium as done in equation (3.54) (but neglecting the dynamic terms), we obtain:

$\forall \underline{v}^h \in \mathcal{U}_h^S(\Omega, 0),\ \forall \underline{\boldsymbol{v}} \in \mathbb{R}^{n_S}$,

$$\underbrace{\mathcal{R}(\Delta \underline{u}_k^h; \underline{v}^h, s_k)}_{\underline{\boldsymbol{v}}^T \underline{\boldsymbol{R}}_{k+1}} = \underbrace{\int_\Omega \underline{\underline{\sigma}}_{k+1} : \underline{\underline{\varepsilon}}(\underline{v}^h)\, d\Omega}_{\underline{\boldsymbol{v}}^T \underline{\boldsymbol{g}}_{k+1}} \underbrace{- \int_\Omega \underline{f}_{k+1} \cdot \underline{v}^h\, d\Omega - \int_{\partial\Omega_N} \underline{f}_{k+1}^N \cdot \underline{v}^h\, dS}_{-\underline{\boldsymbol{v}}^T \underline{\boldsymbol{f}}_{k+1}} \tag{3.61}$$

From the above equation we can define the following discretized residual vector $\underline{\boldsymbol{R}}_{k+1}$:

$$\underline{\boldsymbol{R}}_{k+1} = \underline{\boldsymbol{g}}_{k+1} - \underline{\boldsymbol{f}}_{k+1} \tag{3.62}$$

where the vectors $\underline{\boldsymbol{g}}_{k+1} \in \mathbb{R}^{n_S}$ and $\underline{\boldsymbol{f}}_{k+1} \in \mathbb{R}^{n_S}$ correspond to the internal forces and the vector of external forces respectively.

The main idea of the present APHR method is to reduce the norm of $\underline{\boldsymbol{R}}_{k+1}$ but in an inexpensive way. To do so we consider the following reduced order approximation space:

$$\tilde{\mathcal{R}}_m = \left\{ \underline{u}_m \in \mathcal{U}^S(\Omega, \underline{u}^D) \;\middle|\; \underline{u}_m(\underline{x}, t_{k+1}) = \sum_{i=1}^{m} \underline{w}_i(\underline{x}) a_i, \quad \forall \underline{x} \in \Omega \right\} \tag{3.63}$$

where m is the rank of the decomposition and a_i the reduced state variables of the problem at time t_{k+1}. By using the ROM and projecting it into the equilibrium equation (3.62), we obtain the problem needed to be solved to determine the reduced state variables:

$$\underline{\underline{\boldsymbol{W}}}^T \underline{\boldsymbol{R}}_{k+1} = 0 \tag{3.64}$$

where we recall the use of the matrix $\underline{\underline{\boldsymbol{W}}} \in \mathbb{R}^{n_S} \otimes \mathbb{R}^m$ containing all the reduced space basis given by:

$$\underline{\underline{\boldsymbol{W}}} = \left[\underline{\boldsymbol{w}}_1, \underline{\boldsymbol{w}}_1, \cdots, \underline{\boldsymbol{w}}_m \right]$$

The previous problem requires to performs matrix-vector multiplications, these operations can eventually be very costly if the number of DOFs of the spatial discrete problem is large. In this sense, in order to reduce the cost associated with the computation of the state variables, the APHR method considers a reduced integration domain (RID) for the resolution of problem (3.64) as shown in figure 3.5. This is achieved by introducing a rectangular matrix $\underline{\underline{\boldsymbol{\Pi}}}$ that allows to perform the selection of specific equilibrium equations associated to different elements in space. By introducing an orthogonal condition for this reduction of the equilibrium equations we obtain:

$$\underline{\underline{\boldsymbol{W}}}^T \underline{\underline{\boldsymbol{\Pi}}}^T \underline{\underline{\boldsymbol{\Pi}}}\, \underline{\boldsymbol{R}}_{k+1} = 0 \tag{3.65}$$

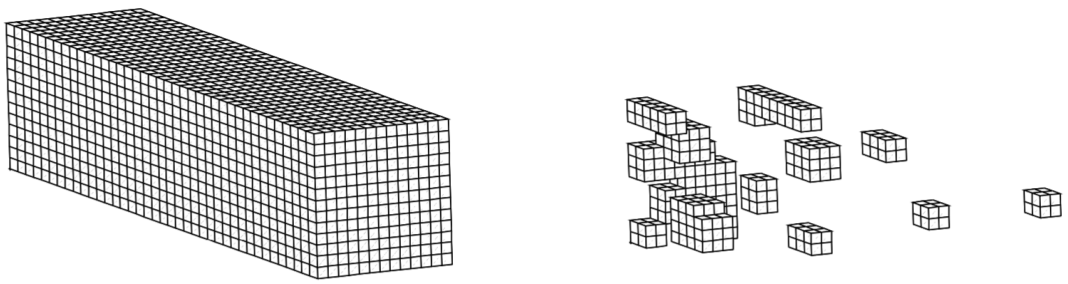


Figure 3.5: Full spatial domain of the reference problem (left) and Reduced Integration Domain (RID) (right) in space considered for the computation of the reduced state variables of the ROM approximation.

Once the state variables are determined, the displacement field is reconstructed over the entire spatial domain simply using the reduced basis already calculated:

$$\underline{\boldsymbol{u}}(t_{k+1}) \approx \sum_{i=1}^{m} \underline{\boldsymbol{w}}_i a_i \quad , \quad a_i \in \mathbb{R} \tag{3.66}$$

On the other hand, the construction of the reduced order basis can be done in many ways [Ryckelynck, 2005, Ryckelynck et al., 2006b], being the following two the principal ones:

1. **Adaptation of the basis functions by a Karhunen-Loève expansion**

   Here a reduced basis is constructed and adapted during the resolution of the nonlinear problem by a Karhunen-Loève expansion of the evolution of reduced state variables.

2. **Construction of a basis by snapshots**.

   A variant, introduced in [Ryckelynck, 2009], exploits the idea of determining a reduced order basis by applying the snapshot POD to the displacement solution of the reference nonlinear problem and also to the local result related to the integration of the constitutive relation, this is, the internal variables. The main idea behind this snapshot POD is to reduce the computational cost for the evaluation of the constitutive relation, this because numerical results show that even if a reduced order model is used to accelerate the resolution of the nonlinear equilibrium equations, this computational time-saving is neglected when treating large discretized problem in space due to the cost associated to the evaluation of the constitutive relation. Therefore, in [Ryckelynck, 2009], in order to reduce this local evaluation of the constitutive relations, the nonlinear material behavior is calculated only on the reduced integration domain (see figure 3.5) and therefore, a reduced basis obtained by snapshots for the internal variables is used to reconstruct the complete field in the whole space domain. This idea is based on the concept of Gappy POD first introduced in [Everson and Sirovich, 1995].

3.4 Non incremental nonlinear solver that use low-rank approximations: the LATIN-PGD framework

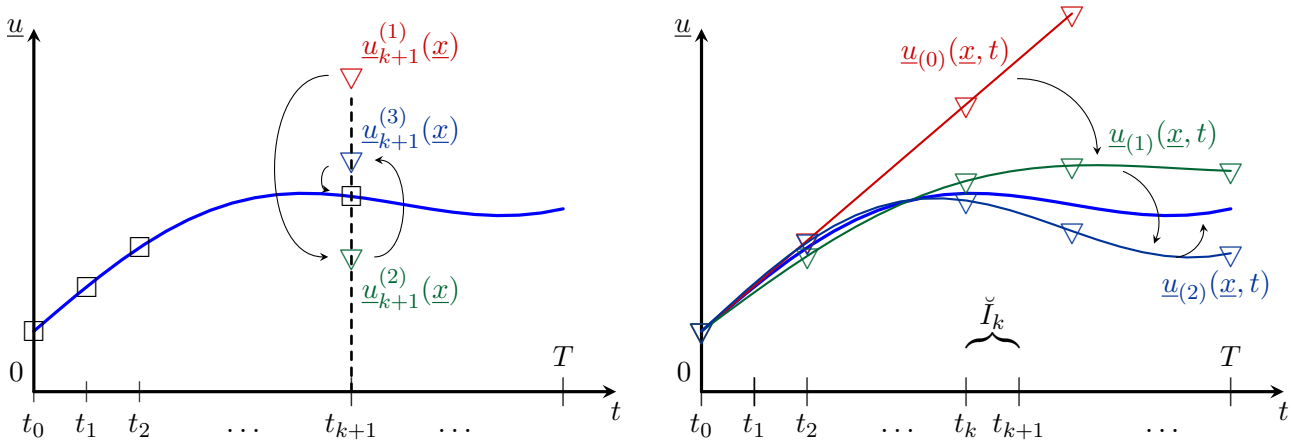
In this section, a global overview of the LATIN-PGD method is introduced. Also a short description of all the different applications of the method developed along the years is presented in the following subsections. A particular emphasis is put on the advantages of using this framework for the fast resolution of nonlinear problems of different kinds.

3.4.1 Introduction

The LATIN method was introduced in [Ladevèze, 1985] and consists in approaching the solution set $\mathcal{S}$ as the limit of a sequence $\mathcal{S}_n$ of iterates defined on the whole space-time domain and containing the solution fields of the reference problem. The solution set is defined as $\mathcal{S} = (\text{dual variables}, \text{primal variables})$. For instance, in the case of elasto-visco-plasticity, one has $\mathcal{S} = \{\underline{\underline{\dot{\varepsilon}}}^p, \underline{\underline{\sigma}}, \underline{\underline{\dot{X}}}, \underline{\underline{Z}}\}$. This solution set is denoted differently from the one introduced for the incremental solver (s), capital notation is chosen to highlight that the sought solution concerns the whole space-time domain.

The main characteristics of the LATIN method is that it is a global nonlinear solver, contrary to classical time-incremental solvers. As illustrated in figure 3.6a, classical incremental solvers consist in minimizing an equilibrium residual iteratively (see section 3.3.1) at current time step t_{k+1} from the knowledge of the solution fields at previous time step t_k. This process is repeated for all the time steps until the whole

temporal domain is solved. On the other hand as presented in figure 3.6b, the LATIN method is also iterative but non-time-incremental; successive corrections to the nonlinear problem are sought on the whole space-time domain at each iteration of the method. As can be seen, the notations related to the LATIN method differ from the classical solver, where the LATIN iterative index is written below. This particular choice is made to better fit the LATIN nomenclature used in the present manuscript. In addition, the first solution of the LATIN method corresponding to the elastic solution is denoted by the index 0, since it is considered outside the iterative process of the LATIN method (the enumeration starts after the solution of a local and global stages).



(a) Classical step-by-step integration.

(b) LATIN strategy.

Figure 3.6: Algorithms for solving nonlinear problems.

In order to define the LATIN framework, the reference problem must be re-formulated, for which, two sets of equations can be distinguished:

1. The equations defining the *admissibility* of the solution $\mathcal{S}$. Two sets of equations in dynamics define admissibility for the solution: the global equilibrium equation over the whole time-space domain and the prescribed boundary conditions on $\partial_D\Omega$. These two equations define an affine subspace $\boldsymbol{A_d}$ of admissible solutions.

2. The equations associated to the nonlinear part of the *constitutive relations*. We denote $\boldsymbol{\Gamma}$ the manifold corresponding to the set of solutions of the constitutive relations.

Those sets of equations are solved implementing an alternation of ***global stages*** and ***local stages***, as explained in the following.

1. ***Initial elastic solution (first solution of the global stage)***:

   The initialization of the method begins with the elastic calculation of the structure on the whole space-time domain, this solution set is denoted $\mathcal{S}_0$. This elastic solution verifies all the boundary conditions, i.e. imposed external forces and displacements. This process is illustrated in the red straight line ($\underline{u}_{(0)}(\underline{x},t)$) of figure 3.6b.

   Once the elastic solution is determined, the iterative process of the LATIN method begins, which consists in the alternate resolution of local and global stages. This procedure is detailed in the following lines.

2. ***Loop iterations***: Iterative resolution between the local and global stages.

   - ***Local stage***: At the beginning of the local stage, a full space-time solution $\mathcal{S}_n$ is available (equal to the elastic admissible solution for the first iteration) but the constitutive relations are not verified. In order to verify the material behavior, an ascent search direction $\mathbb{G}$ is used. This search direction allows the projection of the current solution iterate from the admissible space $\boldsymbol{A_d}$, onto the manifold $\boldsymbol{\Gamma}$ (see Fig. 3.7a). The search direction is defined using a linear operator acting on every space-time integration points. Therefore, the solution $\hat{\mathcal{S}}_{n+1/2}$ that is sought in $\boldsymbol{\Gamma}$ at the local stage implies the resolution of the constitutive relations verifying at the same time the ascent search direction.
   - ***Global stage***: Once the solution $\hat{\mathcal{S}}_{n+1/2}$ of the local stage is computed, one needs to reimpose equilibrium and kinematic admissibility. Introducing then a descent search direction $\mathbb{A}$, the new iterate $\mathcal{S}_{n+1}$ lies at the intersection of two affine spaces (see Fig. 3.7b). The resolution is thus linear but implies the re-imposition of equilibrium on the whole space-time domain. The re-imposition of the equilibrium equation is sought by computing corrective terms which allow to improve the global solution.

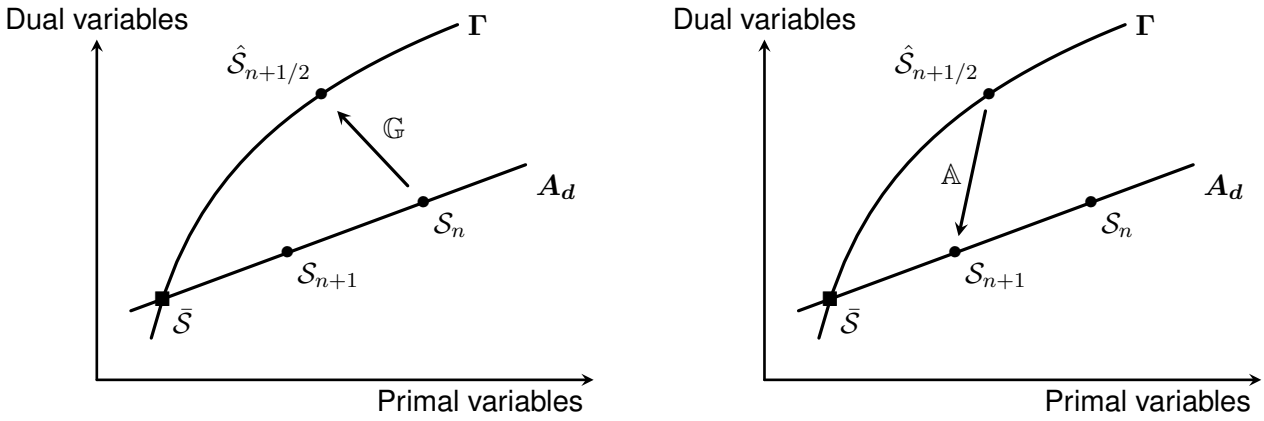
     The global stage can be really numerically expensive due to the full inversion of a space and time discretized problem. But a whole space and time resolution enables the introduction of the model order reduction *Proper Generalized Decomposition* presented on section 3.2.5. This alliance between the LATIN and the PGD greatly improves the numerical performance of the nonlinear solver. The corrections will then be sought as a low-rank approximation, i.e. a combination of PGD modes, products of separate variables functions. The global stage is divided into two sub-stages called **enrichment step** and the **preliminary step**, which are explained below.

     – **Enrichment step**: This step consists in a classical calculation of a PGD basis that approximates the global correction. New PGD modes are computed such that a convergence criterion is reached. Let one note that the PGD corrections have to be computed for every solution field of the reference problem (displacement, stress, damage, plastic strain tensor, etc.).
     – **Preliminary step**: The Enrichment step is of course mandatory at the beginning of the algorithm when no PGD modes are computed, but if the already computed PGD basis is rich enough to correctly approach the corrections along the iterations, enrichment stage can be skipped. Thereby, prior to the calculation of a new PGD couple an additionally step is performed, consisting in fixing all the spatial PGD functions of the problem and actualizing only the temporal functions. This procedure not only decreases the overall computational cost due to an optimized computation of temporal functions by avoiding the expensive calculation of new spatial PGD functions, but also increases the convergence rate of the method.

The iterative resolution, starting from the elastic solution $\mathcal{S}_0$ can be summarized writing:

$$\mathcal{S}_0 \in \boldsymbol{A_d} \longrightarrow \hat{\mathcal{S}}_{1/2} \in \boldsymbol{\Gamma} ... \longrightarrow \hat{\mathcal{S}}_{n+1/2} \in \boldsymbol{\Gamma} \longrightarrow \mathcal{S}_{n+1} \in \boldsymbol{A_d} ... \longrightarrow \bar{\mathcal{S}} \in \boldsymbol{A_d} \cap \boldsymbol{\Gamma}. \tag{3.67}$$

where the distance between the two spaces $\boldsymbol{\Gamma}$ and $\boldsymbol{A_d}$ is decreased with the use of the search directions $\mathbb{G}$ and $\mathbb{A}$ at each iteration of the LATIN method. The solution $\bar{\mathcal{S}}$ over the whole space-time domain then lies at the intersection of those two subspaces.



(a) Local stage: calculation of the local solution of space $\mathbf{\Gamma}$.

(b) Global stage: calculation of the global solution of space $\boldsymbol{A_d}$.

Figure 3.7: Local and global stages; iterative steps of the LATIN method.

During the last few years a wide range of developments have been applied to the LATIN method in order to solve the most challenging problems. In order to give an overview of the main variants and applications of the method, the following sections introduce the areas where the LATIN method has been successfully applied.

3.4.2 Resolution of parametrized nonlinear solid mechanics problems

The idea of applying the LATIN-PGD to solve nonlinear parametric problems resides mainly in the use of the PGD decomposition for the approximation of the linear manifold solution. Its representation as a sum of separate variable functions products leads to the idea of reusing such basis for solving parametrized problems or extending the space-time decomposition to a more general space-time-parameter one. Both cases will be briefly presented in this section.

3.4.2.1 Case of elasto-visco-plastic problems

In this section, an example of the strategy that could be implemented for the fast resolution of parametrized solid mechanics problems considering an elasto-visco-plastic behavior is given. The reader interested in further details is referred to [Néron et al., 2015] and to many related developments as for the case of diffusion problems [Heyberger et al., 2012, Heyberger et al., 2013] or earlier developments in the case of buckling [Boucard and Ladevèze, 1999].

Let one consider a structure for which one wishes to compute the mechanical response for several parameters sets $(\theta_i)_{i=1}^P \in I_\theta$. Assuming that the solution $\mathcal{S}^{i+1}$ (see figure 3.8), associated to manifold $\mathbf{\Gamma}^{i+1}$ is close to the solution $\mathcal{S}^i$, associated to manifold $\mathbf{\Gamma}^i$, if the parameters vectors θ_{i+1} and θ_i describing the manifolds are sufficiently similar, one can define two improvements:

- The solution $\mathcal{S}^{i+1}$ can be initialized to $\mathcal{S}^i$, and not to the initial solution as for the classical LATIN version, with the objective of reducing the number of iterations.
- The PGD basis that have been already computed are preciously kept for being reused when computing $\mathcal{S}^{i+1}$ at global stages; this enables to skip most PGD enrichment sub-stages and perform additional computational time-saving.

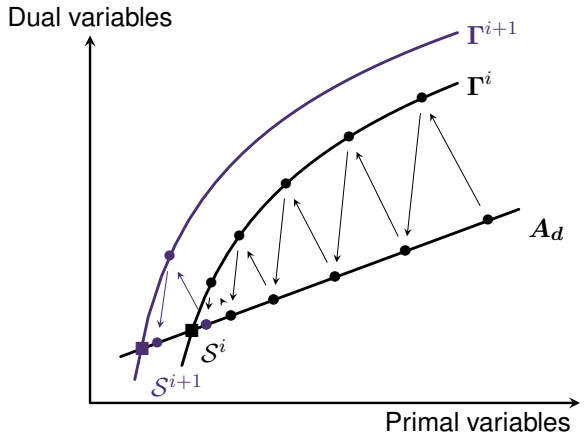
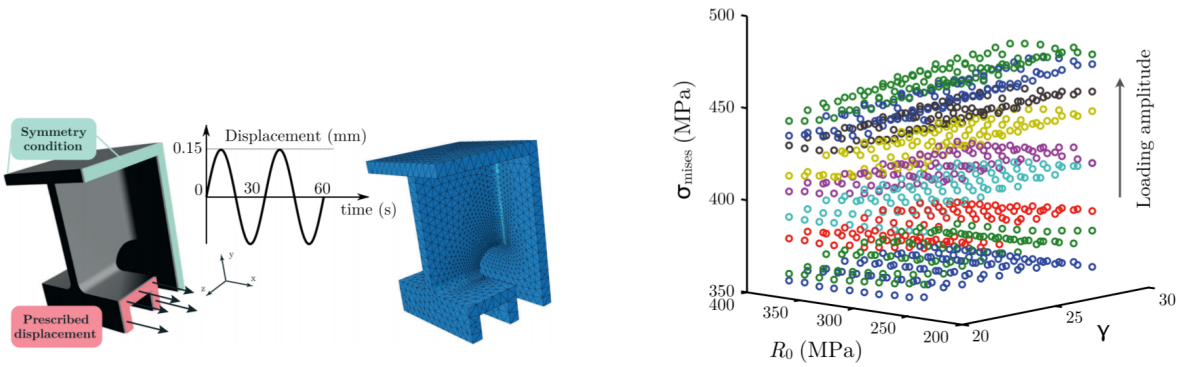


Figure 3.8: Graphical representation of the LATIN method when applying the re-utilization of the PGD basis and optimal initialization.

In [Néron et al., 2015], the reference problem is a quasi-static isothermal evolution of a structure assuming small perturbations, and one of the numerical tests considered consists of a blade with imposed displacements as see in figure 3.9a. For the test studied many material parameters were considered as well as the loading amplitude of the imposed displacement.



(a) Reference blade test with defined boundary conditions.

(b) Maximum von Mises of the parametric study calculated using the LATIN-PGD parametric strategy.

Figure 3.9: Test case and parametric results.

In order to estimate the gain in computational cost of the proposed reuse of PGD basis compared to a classical LATIN-PGD resolution in function of the number of parameters i considered, the following indicator was defined:

$$G^i = \frac{i \times T_1}{T_{i,param}}$$

where T_1 corresponds to the time required for the calculation of the nonlinear solution for a given parametric set and $T_{i,param}$ the whole CPU time when applied the strategy of reusing the reduced basis. This gain was studied as a function of the number of parameters considered, obtaining the figure 3.10.

The last results estimated a gain of 27 when a sufficient number of parameters are considered, which was further verified on numerical tests using the commercial solver ABAQUS. In this test 1000 parameters were considered. This strategy of re-using the PGD basis led to greatly decrease the computational costs, where the following results were obtained:

- 25 days (estimated time) necessary to complete the 1000 resolutions with ABAQUS.
- 18 days (estimated time) necessary to complete the 1000 resolutions with the classical LATIN-PGD.
- Less than 17 h to complete the 1000 resolutions with the new strategy.

From the above results 97 $[\%]$ reduction of the computational time was obtained when compared to the classic incremental method implemented in ABAQUS and a factor of 25 in gain was obtained when compared to the classic LATIN-PGD resolution, verifying the curve of figure 3.10. The results of the maximum von Mises stress as a function of the parameters considered can be see in figure 3.9b.

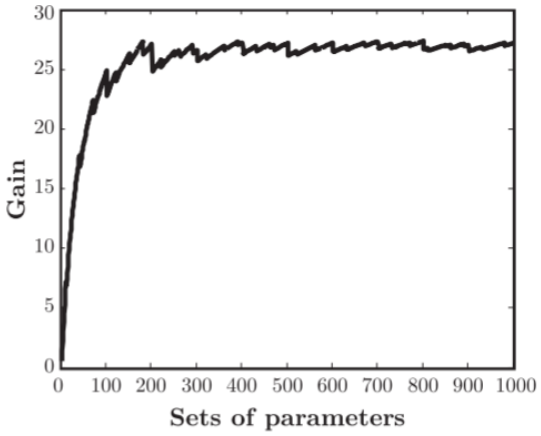


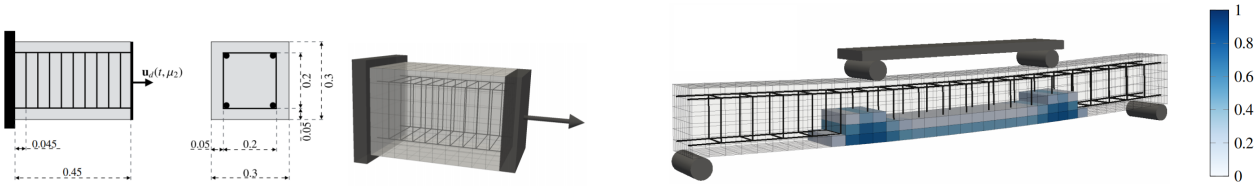
Figure 3.10: Estimated gain G^i by reusing the reduced PGD basis.

3.4.2.2 Case of isotropic damage concrete problems

Another parametrization strategy has been applied to solve parametric isotropic damage in concrete material problems. This work, developed on Vitse's thesis [Vitse, 2016] and presented in [Vitse et al., 2019], consists in extending the space-time decomposition of the PGD to a more general space-time-parameter one:

$$\underline{u}_{m+1}(\underline{x}, t, \theta_1, ..., \theta_{n_P}) = \underline{u}_0(\underline{x}, t, \theta_1, ..., \theta_{n_P}) + \sum_{i=1}^{m+1} \underline{w}_i(\underline{x}) \lambda_i(t) \prod_{j=1}^{n_P} \pi_i^j(\theta_j) \tag{3.68}$$

where $\pi^j(\theta_j)$ corresponds to the PGD function of the parameter $\theta_j \in I_\theta$. This work treated, as usual, a quasi-static isothermal problem assuming small perturbations. For this study, a reinforced-concrete medium was tested in tension (see Fig. 3.11a) and in bending (see Fig. 3.11b) and varying parameters were related to the amplitude of the loading displacement and damage thresholds concrete parameters.



(a) Traction test considered.

(b) Damage field of the bending test.

Figure 3.11: Example of test cases considered in [Vitse, 2016, Vitse et al., 2019].

3.4.3 Treatment of high-cycle fatigue

The global strategy of resolution and the decoupling of the equations due to the PGD space-time decomposition allows to treat also fatigue problems whose resolution using classical incremental approaches are prohibitive. Lets consider a structure submitted to a given loading, that could be, inertial loading, imposed displacements or external forces and lets consider that the temporal content of these loadings are composed of periodic functions whose amplitude varies slowly during the studied time interval $I = [0, T]$. The pioneering works of [Cognard and Ladevèze, 1993, Arzt and Ladevèze, 1994] described a strategy for solving this temporal problem at low numerical cost in the framework of cyclic elasto-visco-plasticity. The resolution of the linear problem on the full interval $I = [0, T]$ was replaced by an approximated resolution on a coarse regularly-spaced discretization called ***macro intervals*** (see Fig. 3.12). Each macro interval is limited at the beginning and at the end by the so called ***nodal cycles***. From the solution known on the nodal cycles, the solution on the full interval $[0, T]$ is interpolated by using shape functions adapted to the problem (linear [Cognard and Ladevèze, 1993] or quadratics [Arzt and Ladevèze, 1994] depending on the evolution of the envelope of the sinusoidal functions) that evolves slowly over the macro intervals.

The improvements described by [Cognard and Ladevèze, 1993] mainly concern the global stage of the LATIN method, where the operators used to compute the temporal PGD functions are interpolated over $I = [0, T]$.

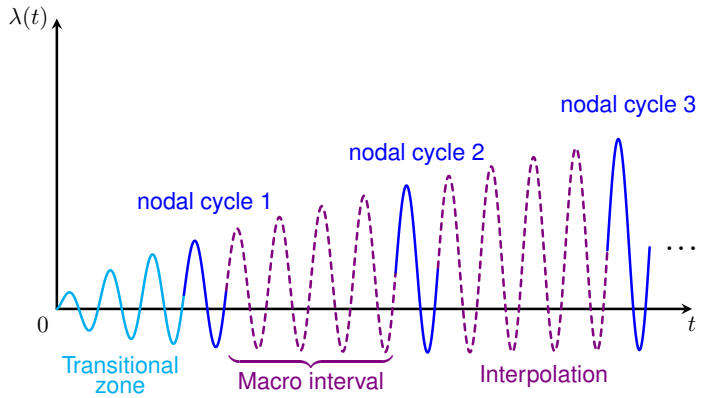


Figure 3.12: Nodal cycles and interpolation strategy introduced in [Cognard and Ladevèze, 1993].

The main idea consists in computing the operator on each nodal cycle and using an affine approximation for the in-between cycles; the initial conditions of current nodal cycle can thus be retrieved, cycle after cycle, in an inexpensive iterative scheme, from the previous nodal cycle's final condition. For more details see also [Ladevèze, 1996, Ladevèze, 1999]. More recently, similar approaches have been introduced for the treatment of elasto-visco-plasticity with damage [Bhattacharyya et al., 2018a, Bhattacharyya et al., 2018b, Bhattacharyya et al., 2018c, Bhattacharyya et al., 2019]; these approaches also make use of the idea of interpolation between nodal cycles, but whose interpolation is applied to the converged solution of the problem rather than the computation of the time functions in the global stage.

3.4.4 Domain decomposition, parallel resolution and multiscale strategy

A special extension of LATIN-PGD has been developed to take advantage of the parallel architecture of current machines and to distribute the computational load among several processors.

The main idea lies in splitting the entire domain into subdomains, each of which having its own constitutive relations and equilibrium equations, and interfaces by the means of which each spatial subdomain

communicates with its neighbors (see figure 3.13). Under these considerations, the resolution of each subdomain can be distributed to the different cores of the hardware architecture.

The novelty of this idea is that the interfaces are considered as mechanical entities and not just mathematical surfaces, in this sense, in addition to the constitutive relation related to the material behavior, there is also the behavior of the interfaces, which can be perfect contact, frictional contact or a more general and complex nonlinear behavior.

The ideas of this strategy were first introduced in [Ladevèze and Lorong, 1991, Champaney et al., 1997, Dureisseix and Ladevèze, 1998], however many developments and improvements have been done in the field since then. For instance, the introduction of a micro-macro spatial representation of the unknowns in [Ladevèze and Dureisseix, 2000] due to the degradation of the convergence's rate of the original idea when the number of subdomains increases (for more details see [Ladevèze, 1996, Dureisseix, 1997, Ladevèze, 1999]).

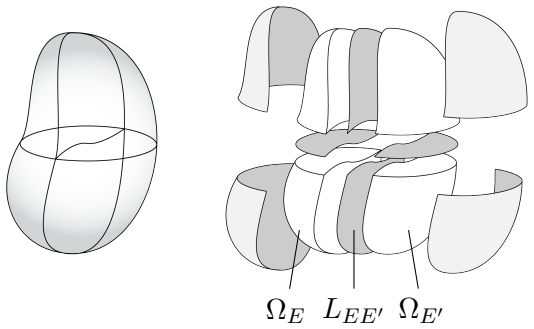


Figure 3.13: Domain decomposition strategy in space, where Ω_E and $\Omega_{E'}$ denote different subdomains that share a common interface given by $L_{EE'}$.

The aforementioned developments allowed a parallelization for the resolution of the spatial problem, however in time a serial resolution still had to be performed. To avoid this problem and to achieve a full spatio-temporal parallel resolution, a spatio-temporal homogenization technique was included to the multiscale strategy [Ladevèze and Nouy, 2002, Ladevèze and Nouy, 2003], which basically consists in extending the idea of micro-macro representation to the temporal domain by defining a "macro" and "micro" temporal discretization. By defining the coarse interval of a macro temporal element "i" as $I_i = [T_{i-1}, T_i]$ (such as $T_0 = 0$ and $T_N = T$ with N the total macro elements in time), the temporal domain is discretized as seen in figure 3.14.

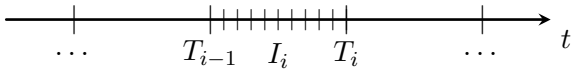


Figure 3.14: Interval of a macro element "i", which in turn is divided into a fine discretization.

The technique consists in approximating with different strategies the macro temporal evolution from the micro evolution, achieving a serial resolution for the temporal macro problem (defined on the coarse discretization) and a parallel and independent resolution for the temporal micro problem (defined in all temporal subdomains). The problem is iteratively solved until continuity and convergence is reached.

These works were further improved in [Ladevèze et al., 2007, Passieux et al., 2008, Passieux et al., 2010, Ladevèze et al., 2010a, Ladevèze et al., 2010b]. For more information about this space-time homogenization technique see [Relun et al., 2011] where a detailed example is given and the following thesis works [Néron, 2004, Passieux, 2008].

This domain decomposition strategy has been successfully applied to the delamination of 3D composite structures [Kerfriden, 2008, Kerfriden et al., 2009, Saavedra Redlich, 2012], to multiphysics problems [Dureisseix et al., 2003, Néron and Dureisseix, 2008], the treatment of multiscale problems [Alart and Dureisseix, 2008, Cremonesi et al., 2013], contacts problems [Oumaziz et al., 2017, Oumaziz et al., 2018] or more recently in a deep analysis of the LATIN multiscale [Oumaziz et al., 2021].

3.4.5 Actual limitations of the method

Despite the wide range of applications solved using the LATIN-PGD strategy and the many developments brought towards the years, proving through examples that drastic computational time-saving can be obtained when solving nonlinear problems, there still exist some limitations:

1. **Extension for the treatment of dynamics problems using the PGD**:

   All the developments of the LATIN-PGD alliance have been done under quasi-static conditions, letting apart dynamic problems which is the very concern of earthquake engineering that this doctoral work aims at developing tools for.

2. **Complexity of the input excitation and duration in time**:

   The external excitation considered in most previous LATIN-applications is of short duration (few seconds) except in the case of fatigue problems. The case of fatigue problems (mentioned in section 3.4.3) is really particular because many assumption are made on the excitation and solution itself, like periodicity that must be verified on the whole duration in time.

   This means that on the one hand, if excitation is complex, better be of short duration, and on the other hand, treating long duration excitation is possible, but with a really small complexity of the signal input. Those limitations naturally brought us to developing a more suitable approach for treating complex excitation that could be of long duration while keeping, at the same time, a low computational cost.

3.5 Conclusions

In the present chapter, an overview of model order reduction strategies was given with a particular emphasis on the way of introducing MOR techniques for solving nonlinear problems. For linear problems, approaches can be classified as *a posteriori* and *a priori* model order reduction:

- ***A posteriori***: this first group is built around the Proper Orthogonal Decomposition (POD), also called Principal Component Analysis or Singular Value Decomposition. It is a method that builds a rough approximation of the range of a reference problem by using the results of this for certain specific points of the time- (for evolutionary problems) or parametric- intervals called snapshots. Due mainly to the fact that the results of reference calculations are needed for its construction such methodologies are called *a posteriori*.
- ***A priori***: in this second group we find the Proper Generalized Decomposition (PGD), which is a method that does not need to know the final results of a problem to generate a reduced basis, but only the linear operators of the problem.

The Certified Reduced Basis Method (CRBM) is also presented, which provides a rigorous mathematical basis for solving evolutionary or parametric problems. In this method the reduced basis is constructed

by using the results on certain and carefully selected snapshots of the parametric domain of the system by employing error estimators in an offline stage, and then approximate the results for any value of the parametric domain in an online stage using Galerkin's projection of the previously calculated basis. The interesting thing about this method is that it delivers low-cost computational expressions for error control in the online stage of the method, making it especially attractive for real-time simulations.

Additionally, nonlinear resolution methods in solid mechanics were presented, which can be classified into two groups, incremental and non incremental solvers:

- **Incremental solver**: Within the incremental classification we have the Newton-Raphson method, which consists in linearizing the equilibrium equations around current time-step.
- **Non incremental or Global solver**: Within the category of non incremental solver we find the LArge Time INcrement (LATIN) method, which allows the linearization of the nonlinear problem in the entire spatial and temporal domain.

Within the classification of methods that solve the nonlinear reference problem in an incremental way, we find again the CRBM framework, where all the developments of the method can be applied to the linearized problem at each iteration of the Newton-Raphson method. In addition, we find the A Priori Hyper-Reduction method (APHR) which solves the nonlinear problem using a reduced basis built on-the-fly as the problem is solved or can be calculated applying the POD on some snapshots. The novelty of the method consists in reducing the spatial domain in where the problem is solved, where only certain number of finite elements of the domain are used to calculate the state variables associated with each reduced basis. Both methods reduce the costs associated with the incremental resolution of nonlinear solid mechanics problems by using reduced basis constructed *a posteriori*, this is, the reduced basis is constructed from the known solution over some snapshots.

On the other hand, the LATIN method allows to linearize the equilibrium equations on the whole space-time domain, allowing to naturally exploit the PGD model reduction method. The LATIN method together with PGD has shown great interest in the last years due to its robustness and has shown great reduction of the computational time in a great variety of problems like parametric studies, fatigue problems, damage of structures like concrete, multiscale problems, parallel resolution, etc.

However, as presented in section 3.4.5 some limitations remain, these include the resolution of dynamics problems by using the LATIN-PGD and the development of dedicated strategies for the fast resolution of problems subjected to complex and long duration signals excitation. These limitations are, in fact, the main points to be solved in the context of earthquake engineering problems, which is the main problem addressed in this thesis work. In this sense, the adaptation of the LATIN-PGD methodology to dynamics is presented in the next chapter 4, in addition, the different developments devoted to improve the performance of the LATIN-PGD method when dealing with long duration and complex excitations are detailed in the chapters 5, 6 and 7, which consists in the introduction of a new temporal multiscale approximation, a hyper-reduction strategy for the fast evaluation of the constitutive relations and a new temporal parallel approach respectively.

Chapter 4

The LATIN-PGD method extended to low-frequency dynamics problems

This chapter presents one of the central contributions of this doctoral work, which consists of the adaptation of the LATIN-PGD framework for solving nonlinear low frequency dynamics problems. The admissibility space is modified to include inertial forces and the computation of the time-space PGD modes at the global stage is rewritten as a minimization problem. It is particularly detailed how this minimization problem is solved using a fixed point strategy implemented in a mixed approach, adopting a Galerkin-like strategy for computing the space modes and a classical residual minimization strategy involving Time-Discontinuous Galerkin for the temporal modes, where a novel incremental resolution strategy is used. The derived approach is carried out for solving nonlinear dynamic problems in the low-frequency range and this involving two different kind of constitutive relations: quasi-brittle damage behavior typical of concrete and elasto-visco-plasticity typical of steel-reinforcements. Numerical examples involving the two considered nonlinear laws are carried out, comparing the LATIN-PGD against the classic Newton-Raphson solver.

Contents

4.1 Introduction **55**
4.2 Initial elastic solution **57**
4.3 Isotropic damage modeling for concrete material **61**
4.4 Elasto-visco-plastic material **71**
4.5 Synthesis on the LATIN-PGD solver **86**
4.6 Numerical examples **88**
4.7 Conclusions **98**

4.1 Introduction

The LATIN method introduced in chapter 3 has shown its performances when dealing with nonlinear solid mechanics problems. The method consists in adding successive corrections to an initial kinematically admissible elastic solution $\mathcal{S}_0$. Every solution field composing the solution $\mathcal{S}$ of the reference problem is corrected on the whole space-time domain in a succession of so-called *local* and *global stages* (see figure 4.1).

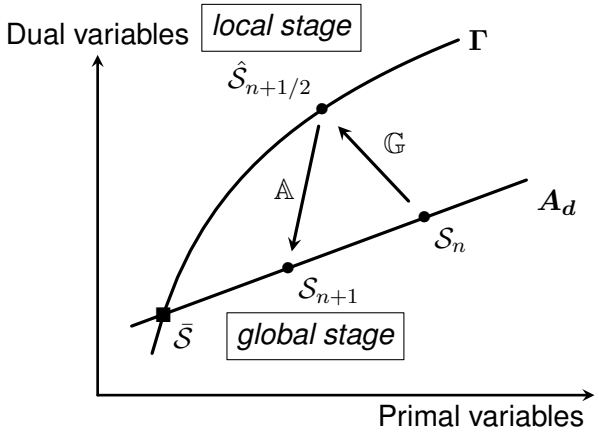


Figure 4.1: LATIN iterative resolution.

The dynamic reference problem introduced in chapter 2, is therefore re-written by introducing:

- An affine space $\boldsymbol{A_d}$, called *admissibility space*, on which initial conditions, kinematic and equilibrium equations are verified, the global stage quantities $\mathcal{S}$ belong to this admissibility space.

- A manifold $\boldsymbol{\Gamma}$ where the nonlinear part of the constitutive relations are verified, and its calculation is done at the local stage where the solution set $\hat{\mathcal{S}}$ is determined.

The jumps from one space to the other are handled by means of so-called *search directions* $\mathbb{A}$ and $\mathbb{G}$, that are very important parameters of the method and allow the alternate resolution of the global and local stages. The local stage consists in the evaluation of the constitutive relation locally on every integration point of the space-time domain, while on the global stage a linear problem must be solved on the whole space-time domain for re-applying admissibility. The linear problem of the global stage is solved by employing a low-rank PGD approximation which brings the LATIN solver its efficiency, and the construction of this decomposition is separated on two main steps, the *enrichment* and the *preliminary* steps. Since we approximate the global quantities $\mathcal{S}$ by a low-rank approximation, on the enrichment step a new mode is added to this decomposition, increasing by consequence its rank, while on the preliminary step the rank is fixed and the temporal PGD functions are updated. A synthetic flowchart of the LATIN methodology is given in the diagram of figure 4.2 where the sub-steps associated to the PGD approximation at global stage are detailed. From this diagram a very important detail must be noticed related to the indexes n and m, which correspond to the LATIN iteration and number of modes of the PGD approximation respectively, which are not necessarily equal. To better distinguish both indexes lets consider the total displacement field at the global stage at iteration n of the LATIN method which is approximated by m PGD terms plus

the initial elastic solution $\underline{u}_0(\underline{x}, t)$:

$$\underline{u}_n(\underline{x}, t) = \underbrace{\sum_{i=1}^{m} \bar{\underline{u}}_i(\underline{x})\lambda_i(t)}_{\text{PGD terms}} + \underline{u}_0(\underline{x}, t) \tag{4.1}$$

The above expression is valid because LATIN iterations increase after the resolution of both local and global stages as shown in the diagram 4.2, however, the number of modes only increases at the enrichment step. On the preliminary step, only an actualization of the temporal PGD functions is done, maintaining the same number of PGD modes. For the above reasons, the LATIN iterations are eventually larger than the PGD modes determined. The enrichment and preliminary steps are carried out in order to obtain the best approximation of the PGD so as to minimize the distance between the local and global solution quantities after each LATIN iteration. The convergence of the LATIN solver is achieved when the distance between the two solutions sets $\mathcal{S}$ and $\hat{\mathcal{S}}$ is less than a given threshold [Ladevèze, 1996, Ladevèze, 1999].

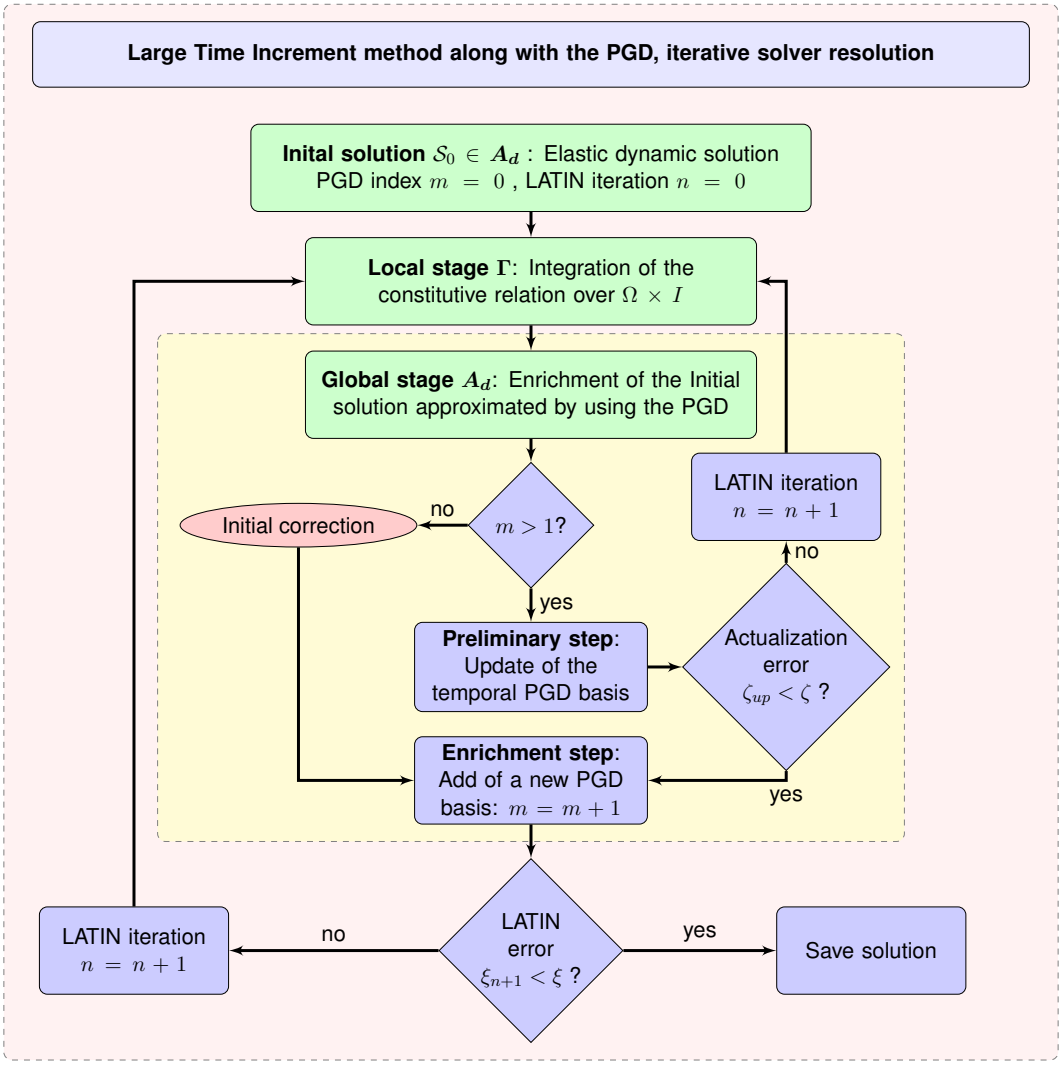


Figure 4.2: LATIN-PGD strategy.

The large application fields of the LATIN method (see chapter 3) in nonlinear mechanics, including param-

eter dependency, makes it an excellent candidate for answering to the virtual charts problematic proposed by seismic risk assessment and described in the introduction chapter 1. However, the application of the LATIN method together with PGD has only been developed for problems under quasi-static conditions. Therefore one of the objective of the current chapter is to adapt the methodology described from quasi-static to the framework of nonlinear low frequency dynamics problems. Another novelty introduced in this chapter is the use of the Time Discontinuous Galerkin Method (TDGM) to obtain the time functions of the PGD decomposition incrementally over the whole time interval for both enrichment and preliminary steps. This allows an efficient resolution of the time functions since the discretized operators to be inverted in each discontinuous time interval are of reduced and constant size, which causes that the computational cost associated with their resolution grows linearly with the size of the temporal domain considered. This property is of particular interest in seismic engineering applications, where the excitation could be of relatively long duration. The idea of using TDGM in the LATIN-PGD solver is not new, in fact it has been extensively used in previous works [Nouy, 2003, Néron, 2004, Gupta, 2005, Passieux, 2008, Nachar, 2019], however, in those works the TDGM was never applied to incrementally solve the time functions while minimizing complicated functionals as will be seen in successive sections.

Every block described in the flow chart of figure 4.2 will be detailed in the next sections, starting from the computation of the initial elastic solution (sec. 4.2), to the description of the *local* and *global* stages.

Let us recall that our ambition is to deal with two kinds of material nonlinearities, naming quasi-brittle damaging concrete and elasto-visco-plastic steel. Unfortunately, the application of the LATIN strategy to those two kinds of nonlinearities requires dedicated adaptations, especially when computing the PGD corrections. For the sake of completeness, the derived methodology will be described in two similar and barely independent sections dedicated to concrete material on the one hand (sec. 4.3) and elasto-visco-plasticity on the other hand (sec. 4.4).

The methodology will be exemplified by a numerical example consisting of a 3D simply supported beam subjected to imposed displacements. For this beam, the material behaviors described above are considered separately.

4.2 Initial elastic solution

The LATIN method is initialized by determining the first solution of the admissibility space $\mathcal{S}_0 \in \boldsymbol{A_d}$, which corresponds to the elastic solution of the reference problem, as can be seen in figure 4.3.

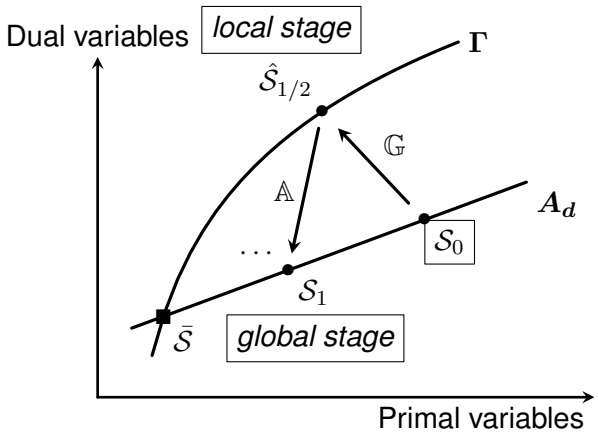


Figure 4.3: Initialization of the solver by the elastic solution $\mathcal{S}_0$.

This solution includes the elastic displacement, strain and stress:

$$\mathcal{S}_0 = \{\underline{u}_0, \underline{\underline{\varepsilon}}_0, \underline{\underline{\sigma}}_0\} \tag{4.2}$$

In order to determine the elastic solution, the space domain is first discretized by employing the Finite Element Method (FEM), where the spatial nodal unknowns are determined at a given time by the following semi-discretized problem:

$$\underline{\underline{\boldsymbol{M}}}\,\underline{\ddot{\boldsymbol{u}}}_0(t) + \underline{\underline{\boldsymbol{D}}}\,\underline{\dot{\boldsymbol{u}}}_0(t) + \underline{\underline{\boldsymbol{K}}}\,\underline{\boldsymbol{u}}_0(t) = \underline{\boldsymbol{f}}(t) \tag{4.3}$$

with $\underline{\underline{\boldsymbol{M}}}$, $\underline{\underline{\boldsymbol{D}}}$ and $\underline{\underline{\boldsymbol{K}}}$ the mass, damping and stiffness matrices respectively; the vector $\underline{\boldsymbol{f}}(t)$ contains the Neumann load conditions and body forces.

As exposed in previous chapters, in the present thesis the TDGM is chosen for the simulation of the time evolution, when this discontinuous strategy in time is applied to the semi-discretized equation (4.3) we obtain the following discretized equations defined on the discontinuous interval $\breve{I}_k$, with "k" a temporal element (see chapter 2):

$$\begin{aligned}\left[\underline{\underline{\boldsymbol{M}}} \otimes (\underline{\underline{\boldsymbol{Q}}}_k^{12} + \underline{\underline{\boldsymbol{P}}}_k^{11}) + \underline{\underline{\boldsymbol{D}}} \otimes \underline{\underline{\boldsymbol{Q}}}_k^{11} + \underline{\underline{\boldsymbol{K}}} \otimes (\underline{\underline{\boldsymbol{Q}}}_k^{10} + \underline{\underline{\boldsymbol{P}}}_k^{00})\right] : \underline{\underline{\mathbf{u}}}_0^{[k]} = \\ (\underline{\underline{\boldsymbol{I}}} \otimes \underline{\underline{\boldsymbol{Q}}}_k^{10}) : \underline{\underline{\mathbf{f}}}^{[k]} + \left[\underline{\underline{\boldsymbol{M}}} \otimes \underline{\underline{\boldsymbol{R}}}_k^{11} + \underline{\underline{\boldsymbol{K}}} \otimes \underline{\underline{\boldsymbol{R}}}_k^{00}\right] : \underline{\underline{\mathbf{u}}}_0^{[k-1]}\end{aligned} \tag{4.4}$$

The discrete solution $\underline{\underline{\mathbf{u}}}_0^{[1]}$ over the initial element $\breve{I}_1$ in turn verifies:

$$\begin{aligned}\left[\underline{\underline{\boldsymbol{M}}} \otimes (\underline{\underline{\boldsymbol{Q}}}_1^{12} + \underline{\underline{\boldsymbol{P}}}_1^{11}) + \underline{\underline{\boldsymbol{D}}} \otimes \underline{\underline{\boldsymbol{Q}}}_1^{11} + \underline{\underline{\boldsymbol{K}}} \otimes (\underline{\underline{\boldsymbol{Q}}}_1^{10} + \underline{\underline{\boldsymbol{P}}}_1^{00})\right] : \underline{\underline{\mathbf{u}}}_0^{[1]} = \\ (\underline{\underline{\boldsymbol{I}}} \otimes \underline{\underline{\boldsymbol{Q}}}_1^{10}) : \underline{\underline{\mathbf{f}}}^{[1]} + (\underline{\underline{\boldsymbol{M}}}.\underline{\dot{\boldsymbol{u}}}_{in}) \otimes \underline{\underline{\boldsymbol{P}}}_1^{1} + (\underline{\underline{\boldsymbol{K}}}.\underline{\boldsymbol{u}}_{in}) \otimes \underline{\underline{\boldsymbol{P}}}_1^{0}\end{aligned} \tag{4.5}$$

where we recall the following operations employed:

$$(\underline{\underline{\boldsymbol{A}}} \otimes \underline{\underline{\boldsymbol{B}}}) : \underline{\underline{\boldsymbol{C}}} = \underline{\underline{\boldsymbol{D}}} \Longleftrightarrow \sum_{j,l} A_{ij} B_{kl} C_{jl} = D_{ik}$$

The above equations allow to determine the dynamic response of the elastic problem incrementally in time. However, its resolution could be really expensive due to the need of inverting matrices of size $n_S \times 4$, with n_S the DOFs in space and 4 the number of unknowns on a temporal element by using cubic Hermite shape functions (see chapter 2). In the following, an approximate way of solving these equations is presented, which resolves this limitation through the use of model order reductions.

4.2.1 Approximation of the elastic dynamic solution

The strategy that will be shown in this section corresponds to a well known modal base approximation of the elastic solution in dynamics (for more details see e.g [Clough and Penzien, 2003]). The approximation consists in separating the total elastic solution as the sum of two terms, a quasi-static term and dynamic one. The quasi-static term is approximated by using the classic PGD approach while the dynamic term is approximated exploiting the modal base of the system. Therefore the elastic solution is written as:

$$\underline{u}_0(\underline{x}, t) = \underbrace{\underline{u}_{0,q}(\underline{x}, t)}_{\text{quasi-static}} + \underbrace{\underline{u}_{0,d}(\underline{x}, t)}_{\text{dynamic}} \tag{4.6}$$

The quasi-static solution verifies $\underline{u}_{0,q} \in \mathcal{U}^S(\Omega, \underline{u}^D)$ and the dynamic term $\underline{u}_{0,d} \in \mathcal{U}^S(\Omega, 0)$. Following [Clough and Penzien, 2003], the semi-discretized quasi-static solution is calculated such as it verifies the equilibrium equation in where the dynamics terms are neglected, that is:

$$\underline{\underline{\boldsymbol{K}}}\,\underline{\boldsymbol{u}}_{0,q}(t) = \underline{\boldsymbol{f}}(t) \tag{4.7}$$

by introducing approximation (4.6) into (4.3) we obtain:

$$\underline{\underline{\boldsymbol{M}}}\left(\underline{\ddot{\boldsymbol{u}}}_{0,q}(t)+\underline{\ddot{\boldsymbol{u}}}_{0,d}(t)\right)+\underline{\underline{\boldsymbol{D}}}\left(\underline{\dot{\boldsymbol{u}}}_{0,d}(t)+\underline{\dot{\boldsymbol{u}}}_{0,q}(t)\right)+\underline{\underline{\boldsymbol{K}}}\left(\underline{\boldsymbol{u}}_{0,d}(t)+\underline{\boldsymbol{u}}_{0,q}(t)\right)=\underline{\boldsymbol{f}}(t) \tag{4.8}$$

and by using (4.7) we finally obtain the equation related to the dynamic term:

$$\underline{\underline{\boldsymbol{M}}}\,\underline{\ddot{\boldsymbol{u}}}_{0,d}(t)+\underline{\underline{\boldsymbol{D}}}\,\underline{\dot{\boldsymbol{u}}}_{0,d}(t)+\underline{\underline{\boldsymbol{K}}}\,\underline{\boldsymbol{u}}_{0,d}(t)=-\underline{\underline{\boldsymbol{M}}}\,\underline{\ddot{\boldsymbol{u}}}_{0,q}(t)-\underline{\underline{\boldsymbol{D}}}\,\underline{\dot{\boldsymbol{u}}}_{0,q}(t)=\underline{\boldsymbol{f}}_d(t) \tag{4.9}$$

Both quasi-static and dynamic terms are calculated by discretizing the temporal domain, the discretization corresponding to the quasi-static term is given as:

$$\left[\underline{\underline{\boldsymbol{K}}}\otimes(\underline{\underline{\boldsymbol{Q}}}_k^{10}+\underline{\underline{\boldsymbol{P}}}_k^{00})\right]:(\underline{\underline{\mathbf{u}}}_{0,q})^{[k]}=(\underline{\underline{\boldsymbol{I}}}\otimes\underline{\underline{\boldsymbol{Q}}}_1^{10}):\underline{\underline{\mathbf{f}}}^{[k]}+\left[\underline{\underline{\boldsymbol{K}}}\otimes\underline{\underline{\boldsymbol{R}}}_k^{00}\right]:(\underline{\underline{\mathbf{u}}}_{0,q})^{[k-1]} \tag{4.10}$$

and the space-time discretization of equation (4.9) for the calculation of the dynamic term by:

$$\begin{aligned}\left[\underline{\underline{\boldsymbol{M}}}\otimes(\underline{\underline{\boldsymbol{Q}}}_k^{12}+\underline{\underline{\boldsymbol{P}}}_k^{11})+\underline{\underline{\boldsymbol{D}}}\otimes\underline{\underline{\boldsymbol{Q}}}_k^{11}+\underline{\underline{\boldsymbol{K}}}\otimes(\underline{\underline{\boldsymbol{Q}}}_k^{10}+\underline{\underline{\boldsymbol{P}}}_k^{00})\right]:(\underline{\underline{\mathbf{u}}}_{0,d})^{[k]}=\\(\underline{\underline{\boldsymbol{I}}}\otimes\underline{\underline{\boldsymbol{Q}}}_1^{10}):\underline{\underline{\mathbf{f}}}_d^{[k]}+\left[\underline{\underline{\boldsymbol{M}}}\otimes\underline{\underline{\boldsymbol{R}}}_k^{11}+\underline{\underline{\boldsymbol{K}}}\otimes\underline{\underline{\boldsymbol{R}}}_k^{00}\right]:(\underline{\underline{\mathbf{u}}}_{0,d})^{[k-1]}\end{aligned} \tag{4.11}$$

where $\underline{\underline{\mathbf{f}}}_d^{[k]}\in\mathbb{R}^{n_S}\otimes\mathbb{R}^4$ is obtained by discretizing $\underline{\boldsymbol{f}}_d$ (the right hand side of equation (4.9)) in time using FEM. Following the reference problem of this thesis introduced in chapter 2, the initial conditions associated to the quasi-static and to the dynamic solution of the elastic problem are both set equals to zero:

$$(\underline{\underline{\mathbf{u}}}_{0,q})^{[0]}=0$$
$$(\underline{\underline{\mathbf{u}}}_{0,d})^{[0]}=0$$

The separation of the reference elastic problem into a quasi-static solution which verifies the Dirichlet conditions and a dynamic solution which is admissible to zero, allows the use of model order reduction techniques for its fast calculation. The quasi-static solution is approximated by using the PGD, that is:

$$\underline{u}_{0,q}(\underline{x},t)\approx\sum_{i=1}^{m_q}\underline{w}_q(\underline{x})\lambda_q(t) \tag{4.12}$$

The PGD approximation of the quasi-static solution is calculated by using a classic Greedy approach (see section 3.2.5.4). On the other hand the dynamic term is approximated by using a modal base approach, this is:

$$\underline{u}_{0,d}(\underline{x},t)\approx\sum_{i=1}^{m_b}\underline{\phi}_i(\underline{x})\upsilon_i(t) \tag{4.13}$$

where $\underline{\phi}(\underline{x})$ and $\upsilon(t)$ correspond to the spatial and temporal modal base functions and with m_b the quantity of modal basis considered. The spatial functions are determined by computing the eigenvectors associated to the following eigenvalue problem:

$\forall i\in[1,...,m_b]$,

$$\left(\underline{\underline{\boldsymbol{M}}}^{-1}\underline{\underline{\boldsymbol{K}}}\right)\underline{\boldsymbol{\phi}}_i=\omega_i^2\underline{\boldsymbol{\phi}}_i \tag{4.14}$$

where ω_i^2 corresponds to the eigenvalue associated to eigenvector i and such that $\omega_i=2\pi f_i$, with f_i the natural frequency of the system at mode i. On the other hand the temporal functions $\upsilon_i(t)$ are determined by projecting the discretized spatial modal basis functions into the discretized dynamic part equilibrium equation (4.11). To do so, lets introduce the total discretized unknown of the temporal modal functions as:

$$\underline{\underline{\boldsymbol{\Upsilon}}}^{[k]}=\left[\underline{\boldsymbol{\upsilon}}_1^{[k]},\underline{\boldsymbol{\upsilon}}_2^{[k]},\cdots,\underline{\boldsymbol{\upsilon}}_{m_b}^{[k]}\right]^T \tag{4.15}$$

By projecting the spatial modal basis into the dynamic equilibrium equation we obtain:

$$\begin{aligned}\Big[(\underline{\underline{\boldsymbol{\Phi}}}^T\underline{\underline{\boldsymbol{M}}}\,\underline{\underline{\boldsymbol{\Phi}}})\otimes(\underline{\underline{\boldsymbol{Q}}}_k^{12}+\underline{\underline{\boldsymbol{P}}}_k^{11})+(\underline{\underline{\boldsymbol{\Phi}}}^T\underline{\underline{\boldsymbol{D}}}\,\underline{\underline{\boldsymbol{\Phi}}})\otimes\underline{\underline{\boldsymbol{Q}}}_k^{11}+(\underline{\underline{\boldsymbol{\Phi}}}^T\underline{\underline{\boldsymbol{K}}}\,\underline{\underline{\boldsymbol{\Phi}}})\otimes(\underline{\underline{\boldsymbol{Q}}}_k^{10}+\underline{\underline{\boldsymbol{P}}}_k^{00})\Big]:\underline{\underline{\boldsymbol{\Upsilon}}}^{[k]}=\\ \Big[(\underline{\underline{\boldsymbol{\Phi}}}^T\underline{\underline{\boldsymbol{M}}}\,\underline{\underline{\boldsymbol{\Phi}}})\otimes\underline{\underline{\boldsymbol{R}}}_k^{11}+(\underline{\underline{\boldsymbol{\Phi}}}^T\underline{\underline{\boldsymbol{K}}}\,\underline{\underline{\boldsymbol{\Phi}}})\otimes\underline{\underline{\boldsymbol{R}}}_k^{00}\Big]:\underline{\underline{\boldsymbol{\Upsilon}}}^{[k-1]}+(\underline{\underline{\boldsymbol{I}}}\otimes\underline{\underline{\boldsymbol{Q}}}_k^{10}):(\underline{\underline{\boldsymbol{\Phi}}}^T\underline{\underline{\mathbf{f}}}_d^{[k]})\end{aligned}\tag{4.16}$$

with the equation associated with the first time element:

$$\begin{aligned}\Big[(\underline{\underline{\boldsymbol{\Phi}}}^T\underline{\underline{\boldsymbol{M}}}\,\underline{\underline{\boldsymbol{\Phi}}})\otimes(\underline{\underline{\boldsymbol{Q}}}_1^{12}+\underline{\underline{\boldsymbol{P}}}_1^{11})+(\underline{\underline{\boldsymbol{\Phi}}}^T\underline{\underline{\boldsymbol{D}}}\,\underline{\underline{\boldsymbol{\Phi}}})\otimes\underline{\underline{\boldsymbol{Q}}}_1^{11}+(\underline{\underline{\boldsymbol{\Phi}}}^T\underline{\underline{\boldsymbol{K}}}\,\underline{\underline{\boldsymbol{\Phi}}})\otimes(\underline{\underline{\boldsymbol{Q}}}_1^{10}+\underline{\underline{\boldsymbol{P}}}_1^{00})\Big]:\underline{\underline{\boldsymbol{\Upsilon}}}^{[1]}=\\ (\underline{\underline{\boldsymbol{I}}}\otimes\underline{\underline{\boldsymbol{Q}}}_1^{10}):(\underline{\underline{\boldsymbol{\Phi}}}^T\underline{\underline{\mathbf{f}}}_d^{[1]})\end{aligned}\tag{4.17}$$

with $\underline{\underline{\boldsymbol{I}}}\in\mathbb{R}^m\otimes\mathbb{R}^m$ the identity matrix and $\underline{\underline{\boldsymbol{\Phi}}}$ the matrix that contains the modal basis given by:

$$\underline{\underline{\boldsymbol{\Phi}}}=\Big[\underline{\boldsymbol{\phi}}_1,\underline{\boldsymbol{\phi}}_2,\cdots,\underline{\boldsymbol{\phi}}_{m_b}\Big]\tag{4.18}$$

The modal basis have the interesting property of being orthogonal to the mass and stiffness matrices, but might not be the case for the damping matrix. In either case, the damping is assumed to be diagonal in the modal basis (small damping ratios) and the decoupled equations are written as follows:

$\forall i\in[1,...,m_b]$,

$$\Big[m_i(\underline{\underline{\boldsymbol{Q}}}_k^{12}+\underline{\underline{\boldsymbol{P}}}_k^{11})+d_i\underline{\underline{\boldsymbol{Q}}}_k^{11}+k_i(\underline{\underline{\boldsymbol{Q}}}_k^{10}+\underline{\underline{\boldsymbol{P}}}_k^{00})\Big]\,\underline{\boldsymbol{v}}_i^{[k]}=\Big[m_i\underline{\underline{\boldsymbol{R}}}_k^{11}+k_i\underline{\underline{\boldsymbol{R}}}_k^{00}\Big]\,\underline{\boldsymbol{v}}_i^{[k-1]}+(\underline{\boldsymbol{f}}_d)_i^{[k]}\tag{4.19}$$

with the modal masses, damping and stiffness scalars:

$\forall i\in[1,...,m_b]$,

$$\begin{aligned}m_i&=\underline{\boldsymbol{\phi}}_i^T\underline{\underline{\boldsymbol{M}}}\,\underline{\boldsymbol{\phi}}_i\\ d_i&=\underline{\boldsymbol{\phi}}_i^T\underline{\underline{\boldsymbol{D}}}\,\underline{\boldsymbol{\phi}}_i\\ k_i&=\underline{\boldsymbol{\phi}}_i^T\underline{\underline{\boldsymbol{K}}}\,\underline{\boldsymbol{\phi}}_i\end{aligned}\tag{4.20}$$

and the total external force as:

$$(\underline{\boldsymbol{f}}_d)_i^{[k]}=\underline{\underline{\boldsymbol{Q}}}_k^{10}(\underline{\boldsymbol{\phi}}_i^T\underline{\underline{\mathbf{f}}}_d^{[k]})^T\tag{4.21}$$

For the sake of simplicity, equation (4.19) can be rewritten as follows:

$\forall i\in[1,...,m_b]$,

$$\underline{\underline{\boldsymbol{Q}}}_i^{[k]}\,\underline{\boldsymbol{v}}_i^{[k]}=\underline{\underline{\boldsymbol{\mathcal{R}}}}_i^{[k]}\,\underline{\boldsymbol{v}}_i^{[k-1]}+(\underline{\boldsymbol{f}}_d)_i^{[k]}\tag{4.22}$$

where $\underline{\underline{\boldsymbol{Q}}}_i^{[k]}$ and $\underline{\underline{\boldsymbol{\mathcal{R}}}}_i^{[k]}$ are matrices given by:

$\forall i\in[1,...,m_b]$,

$$\begin{aligned}\underline{\underline{\boldsymbol{Q}}}_i^{[k]}&=\Big[m_i(\underline{\underline{\boldsymbol{Q}}}_k^{12}+\underline{\underline{\boldsymbol{P}}}_k^{11})+d_i\underline{\underline{\boldsymbol{Q}}}_k^{11}+k_i(\underline{\underline{\boldsymbol{Q}}}_k^{10}+\underline{\underline{\boldsymbol{P}}}_k^{00})\Big]\\ \underline{\underline{\boldsymbol{\mathcal{R}}}}_i^{[k]}&=\Big[m_i\underline{\underline{\boldsymbol{R}}}_k^{11}+k_i\underline{\underline{\boldsymbol{R}}}_k^{00}\Big]\end{aligned}\tag{4.23}$$

After the displacement $\underline{u}_0(\underline{x},t)$ is determined, the initial stress and strain tensors are computed as:

$$\underline{\underline{\varepsilon}}_0=\frac{1}{2}\left(\nabla\underline{u}_0+\nabla\underline{u}_0^T\right)\qquad,\qquad\underline{\underline{\sigma}}_0=\mathbb{K}:\underline{\underline{\varepsilon}}_0$$

with $\mathbb{K}$ the Hooke's tensor. Once the initial elastic solution $\mathcal{S}_0$ in dynamics is determined, the iterative process between the local and global stages begins:

$$\mathcal{S}_0 \in \boldsymbol{A_d} \longrightarrow \hat{\mathcal{S}}_{1/2} \in \boldsymbol{\Gamma} \ldots \longrightarrow \hat{\mathcal{S}}_{n+1/2} \in \boldsymbol{\Gamma} \longrightarrow \mathcal{S}_{n+1} \in \boldsymbol{A_d} \ldots \longrightarrow \bar{\mathcal{S}} \in \boldsymbol{A_d} \cap \boldsymbol{\Gamma}. \tag{4.24}$$

which allows to correct the initial solution until the final solution is reached.

In the following section 4.3, this process is explained for the case of isotropic damage in concrete, while the elasto-visco-plasticity case is explained in section 4.4.

4.3 Isotropic damage modeling for concrete material

On the present section the iterative resolution process between the global and local stages for the simulation of isotropic damageable concrete material is introduced. The details about this constitutive relation were presented in chapter 2.

The strategy of resolution presented herein differs from the one introduced in Vitse's work [Vitse, 2016, Vitse et al., 2019]. While the previous work calculated the spatial and temporal functions of the PGD decomposition by a Galerkin orthogonality projection into the equilibrium equations, in the present strategy only the spatial functions are calculated in this way, while the temporal functions are determined by minimizing a constitutive relation error [Ladevèze and Moës, 1998] introduced in the following subsections. The above has shown better convergence properties [Passieux, 2008].

For the following sections, we assume that a global $\mathcal{S}_n$ solution have already been computed.

4.3.1 Local nonlinear stage

Once the linear solution $\mathcal{S}_n \in \boldsymbol{A_d}$ is computed, the local nonlinear solution $\hat{\mathcal{S}}_{n+1/2} \in \boldsymbol{\Gamma}$ is calculated, this solution must verify also an ascent search direction $\mathbb{G}$. This can be resumed as follows:

$$\boxed{\text{Find} \quad \hat{\mathcal{S}}_{n+1/2} \in \boldsymbol{\Gamma} \quad \text{such that} \quad \left(\hat{\mathcal{S}}_{n+1/2} - \mathcal{S}_n\right) \in \mathbb{G}} \tag{4.25}$$

with the local solution given by:

$$\hat{\mathcal{S}}_{n+1/2} = \{\hat{\underline{\underline{\varepsilon}}}_{n+1/2}, \hat{\underline{\underline{\sigma}}}_{n+1/2}, \hat{d}_{n+1/2}\} \tag{4.26}$$

with d the isotropic damage variable (see chapter 2). The local stage process is depicted in figure 4.4.

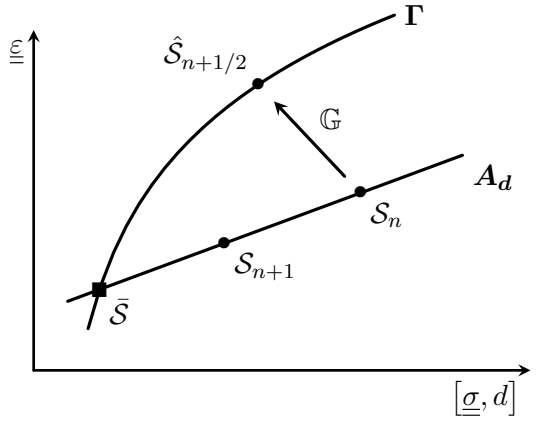


Figure 4.4: Calculation of the local solution of space $\boldsymbol{\Gamma}$.

The ascent search direction is chosen as:

$$\mathbb{G}: \quad \begin{cases} \hat{d}_{n+1/2} - g(\hat{\underline{\underline{\varepsilon}}}_{n+1/2}) = 0 \\ \left[\hat{\underline{\underline{\sigma}}}_{n+1/2} - \underline{\underline{\sigma}}_n\right] + \hat{\mathbb{H}}_\varepsilon : \left[\hat{\underline{\underline{\varepsilon}}}_{n+1/2} - \underline{\underline{\varepsilon}}_n\right] = 0 \end{cases} \tag{4.27}$$

From the above equation we denoted $g : \mathbb{R}^3 \otimes \mathbb{R}^3 \to \mathbb{R}$, the nonlinear function that condense the operations given in chapter 2 for the determination of the damage variable. For simplicity, in the second equation of (4.27) the positive operator $\hat{\mathbb{H}}_\varepsilon$ is chosen as:

$$\hat{\mathbb{H}}_\varepsilon^{-1} = 0 \tag{4.28}$$

which means that the search direction is chosen to be constant. Under this condition, the local strain is simply given by:

$$\hat{\underline{\underline{\varepsilon}}}_{n+1/2} = \underline{\underline{\varepsilon}}_n \tag{4.29}$$

The stress tensor on the other hand is simply determined once the damage is computed, by:

$$\hat{\underline{\underline{\sigma}}}_{n+1/2} = h(\hat{\underline{\underline{\varepsilon}}}_{n+1/2}, \hat{d}_{n+1/2}) \tag{4.30}$$

where $h(\hat{\underline{\underline{\varepsilon}}}_{n+1/2}, \hat{d}_{n+1/2})$ is a function that summarises all the procedures outlined in section 2.3.1 for the simulation of the unilateral effect.

Remark: Note that the damage variable can be calculated locally in space, but not in time, where an ordinary differential equation must be solved as presented in section 2.3.1.

4.3.2 Global linear stage: Equilibrium and compatibility equations

After the resolution of the local stage, where the constitutive relation is evaluated and the solution $\hat{\mathcal{S}}_{n+1/2}$ is calculated, the new global stage solution $\mathcal{S}_{n+1} \in \boldsymbol{A_d}$ at LATIN iteration $n+1$ is computed. This solution must verify the admissibility conditions of the reference problem and at the same time the descent search direction, this is:

$$\boxed{\text{Find} \quad \mathcal{S}_{n+1} \in \boldsymbol{A_d} \quad \text{such that} \quad \left(\mathcal{S}_{n+1} - \hat{\mathcal{S}}_{n+1/2}\right) \in \mathbb{A}} \tag{4.31}$$

with the global stage solution at iteration $n+1$ given by:

$$\mathcal{S}_{n+1} = \{\underline{u}_{n+1}, \underline{\underline{\varepsilon}}_{n+1}, \underline{\underline{\sigma}}_{n+1}, d_{n+1}\} \tag{4.32}$$

The global stage process summarized by equation (4.31) is illustrated in figure 4.5.

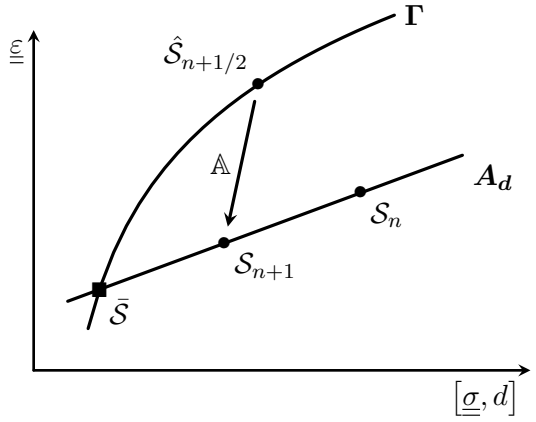


Figure 4.5: Calculation of the global solution of space $\boldsymbol{A_d}$ for the case of isotropic damage.

The descent search direction is given by:

$$\mathbb{A}: \quad \begin{cases} d_{n+1} - \hat{d}_{n+1/2} = 0 \\ \left[\underline{\underline{\sigma}}_{n+1} - \hat{\underline{\underline{\sigma}}}_{n+1/2}\right] - \mathbb{H}_\varepsilon : \left[\underline{\underline{\varepsilon}}_{n+1} - \hat{\underline{\underline{\varepsilon}}}_{n+1/2}\right] = 0 \end{cases} \tag{4.33}$$

from the above expressions we consider the damage quantity at the global stage equals to the one from the local stage:

$$d_{n+1} = \hat{d}_{n+1/2} \tag{4.34}$$

The operator $\mathbb{H}_\varepsilon$ is chosen to be constant and equal to the Hooke's tensor of the concrete material:

$$\mathbb{H}_\varepsilon = \mathbb{K} \tag{4.35}$$

The main objective of the global stage at LATIN iteration $n+1$ consists in determining the solution set $\mathcal{S}_{n+1}$. However, a direct resolution of the global unknowns is generally very expensive, but as exposed in the introductory section the global stage is solved by employing the model reduction technique PGD, that approximates the global stage solution as a low-rank decomposition, which allows to decreases the computational costs. Under this approximation lets assume that m modes have already been computed for approximating the global solution at LATIN iteration n as follows:

$$\begin{aligned} \underline{u}_n(\underline{x},t) &= \sum_{i=1}^{m} \underline{\bar{u}}_i(\underline{x})\lambda_i(t) + \underline{u}_0(\underline{x},t) \\ \underline{\underline{\varepsilon}}_n(\underline{x},t) &= \sum_{i=1}^{m} \underline{\underline{\bar{\varepsilon}}}_i(\underline{x})\lambda_i(t) + \underline{\underline{\varepsilon}}_0(\underline{x},t) \\ \underline{\underline{\sigma}}_n(\underline{x},t) &= \sum_{i=1}^{m} \underline{\underline{\bar{\sigma}}}_i(\underline{x})\lambda_i(t) + \underline{\underline{\sigma}}_0(\underline{x},t) \end{aligned} \tag{4.36}$$

In order to solve the global stage at iteration $n+1$, we seek a corrective term such as the global stage solution can be written as follows:

$$\mathcal{S}_{n+1} = \Delta\mathcal{S}_{n+1} + \mathcal{S}_n \tag{4.37}$$

with the corrective solution set given by:

$$\Delta\mathcal{S}_{n+1} = \{\Delta\underline{u}_{n+1}, \Delta\underline{\underline{\varepsilon}}_{n+1}, \Delta\underline{\underline{\sigma}}_{n+1}\} \tag{4.38}$$

These corrective terms apply only to the stress, strain and displacement. For their determination, the second equation of (4.33) is rewritten as:

$$\Delta\underline{\underline{\sigma}}_{n+1} - \mathbb{H}_\varepsilon : \Delta\underline{\underline{\varepsilon}}_{n+1} + \underline{\underline{\Delta}}_{n+1} = 0 \tag{4.39}$$

with the residual term (considering the result of (4.29)) given as follows:

$$\underline{\underline{\Delta}}_{n+1} = (\underline{\underline{\sigma}}_n - \hat{\underline{\underline{\sigma}}}_{n+1/2}) \tag{4.40}$$

The expression (4.39) expresses the linear constitutive relation given by the descent search direction. It gives an equation in where the corrective terms $\Delta\underline{\underline{\sigma}}_{n+1}$ and $\Delta\underline{\underline{\varepsilon}}_{n+1}$ allow to decrease the distance between the global and local quantities, a key aspect in order to ensure the convergence of the method. In this sense, we define a constitutive relation error J^c [Passieux, 2008] such as:

$$J^c = \left|\left|\left|\Delta\underline{\underline{\sigma}}_{n+1} - \mathbb{H}_\varepsilon : \Delta\underline{\underline{\varepsilon}}_{n+1} + \underline{\underline{\Delta}}_{n+1}\right|\right|\right|^2_{\mathbb{H}_\varepsilon^{-1}} \tag{4.41}$$

with the norm chosen as:

$$|||\cdot|||^2_{\mathbb{H}_\varepsilon^{-1}} = \int_{\Omega\times I} (\cdot) : \mathbb{H}_\varepsilon^{-1} : (\cdot) d\Omega dt \tag{4.42}$$

From the above developments, in order to determine $\mathcal{S}_{n+1}$ we simply need to compute the corrective terms of equation (4.38) over the whole space-time domain such that the constitutive relation error J^c is minimized, while verifying at the same time the equilibrium equations and the admissibility conditions. However, since the global stage quantities are approximated by a low-rank PGD decomposition, the minimization is done by solving two steps: the enrichment and preliminary steps, which are detailed in the following sections.

Remark: After the determination of the corrective solution set $\Delta\mathcal{S}_{n+1}$, a relaxation of the solution is applied in order to ensure the convergence of the method [Ladevèze, 1999], this is simply done as follows:

$$\mathcal{S}_{n+1} \leftarrow (1-\mu)\mathcal{S}_n + \mu\mathcal{S}_{n+1} \tag{4.43}$$

where the relaxation coefficients for the case of isotropic damage is chosen equal to $\mu = 0.4$. This parameter is problem dependent, so it could be potentially different for other kind of applications, where empirical numerical tests must be performed for its optimal determination.

4.3.2.1 Enrichment step: enrichment of the PGD approximation

The enrichment step consists in the calculation of a new PGD mode in order to enrich the PGD approximation for each global quantities. These rank-one modes are determined such that it approximates the corrective term that must be determined at the global stage, in other words, the modes are found as the correction from solution $\mathcal{S}_n$ to $\mathcal{S}_{n+1}$ and are defined as:

$$\begin{aligned}
\Delta\underline{u}_{n+1}(\underline{x},t) &= \underline{u}_{n+1} - \underline{u}_n = \bar{\underline{u}}_{m+1}(\underline{x})\lambda_{m+1}(t) \\
\Delta\underline{\underline{\varepsilon}}_{n+1}(\underline{x},t) &= \underline{\underline{\varepsilon}}_{n+1} - \underline{\underline{\varepsilon}}_n = \bar{\underline{\underline{\varepsilon}}}_{m+1}(\underline{x})\lambda_{m+1}(t) \\
\Delta\underline{\underline{\sigma}}_{n+1}(\underline{x},t) &= \underline{\underline{\sigma}}_{n+1} - \underline{\underline{\sigma}}_n = \bar{\underline{\underline{\sigma}}}_{m+1}(\underline{x})\lambda_{m+1}(t)
\end{aligned} \tag{4.44}$$

The spatial and temporal functions are calculated in an iterative way by using a fixed point strategy. The spatial functions are calculated by verifying the admissibility equations and the temporal functions are calculated by minimizing the constitutive relation error. After each fixed point iteration when a space-time couple of functions are calculated a stagnation error is defined in order to stop the process. To do so, first the spatial functions are normalized with respect to the norm of the deformation tensor, this is, we define the scalar:

$$c_c = \left\| \bar{\underline{\underline{\varepsilon}}}_{m+1} \right\|_\Omega \quad , \quad \|\cdot\|_\Omega = \sqrt{\int_\Omega (\cdot) : (\cdot)\, d\Omega} \tag{4.45}$$

which is used to adapt the spatial and temporal PGD functions:

$$\begin{aligned}
\bar{\underline{u}}_{m+1} &\leftarrow \bar{\underline{u}}_{m+1}/c_c & & \lambda_{m+1} \leftarrow \lambda_{m+1}\, c_c \\
\bar{\underline{\underline{\varepsilon}}}_{m+1} &\leftarrow \bar{\underline{\underline{\varepsilon}}}_{m+1}/c_c \quad , & & \dot{\lambda}_{m+1} \leftarrow \dot{\lambda}_{m+1}\, c_c \\
\bar{\underline{\underline{\sigma}}}_{m+1} &\leftarrow \bar{\underline{\underline{\sigma}}}_{m+1}/c_c & & \ddot{\lambda}_{m+1} \leftarrow \ddot{\lambda}_{m+1}\, c_c
\end{aligned}$$

The stagnation error of the fixed point process is defined as:

$$\zeta_{PGD} = \frac{\left\| |\lambda^{(i)}_{m+1}| - |\lambda^{(i-1)}_{m+1}| \right\|_I}{\left\| |\lambda^{(i)}_{m+1}| + |\lambda^{(i-1)}_{m+1}| \right\|_I} \quad , \quad \|\cdot\|_I = \sqrt{\int_I (\cdot)^2 dt} \tag{4.46}$$

with $|\cdot|$ the absolute value. The functions $\lambda_{m+1}^{(i)}$ and $\lambda_{m+1}^{(i-1)}$ stands for the temporal mode at iteration i and $i-1$ respectively. The fixed point strategy stops when the stagnation indicator reaches a given threshold or a maximum number of iterations of the process is reached. At the end of the process the spatial and temporal functions are kept and are added to the current low-rank approximation.

In this work, the space functions of the decomposition are determined by a Galerkin projection into the equilibrium equation, which allows to easily verify the boundary conditions of the problem, while the temporal function is determined by minimizing the constitutive relation error defined in (4.41). These steps are detailed in the following sections, where, for simplicity, the PGD index is omitted as shown below:

$$\begin{aligned}\Delta\underline{u}_{n+1}(\underline{x},t) &= \underline{\bar{u}}_{m+1}(\underline{x})\lambda_{m+1}(t) = \underline{\bar{u}}(\underline{x})\lambda(t)\\ \Delta\underline{\underline{\varepsilon}}_{n+1}(\underline{x},t) &= \underline{\underline{\bar{\varepsilon}}}_{m+1}(\underline{x})\lambda_{m+1}(t) = \underline{\underline{\bar{\varepsilon}}}(\underline{x})\lambda(t)\\ \Delta\underline{\underline{\sigma}}_{n+1}(\underline{x},t) &= \underline{\underline{\bar{\sigma}}}_{m+1}(\underline{x})\lambda_{m+1}(t) = \underline{\underline{\bar{\sigma}}}(\underline{x})\lambda(t)\end{aligned} \tag{4.47}$$

4.3.2.1.1 Space and time resolution associated to the admissibility problem

In the following lines the space and temporal problems needed to be solved on the enrichment step are detailed; the space problem is first presented, assuming that the temporal functions are known, and later the temporal problem is solved using the result of the space problem. This process is applied in an iterative way until the final PGD mode is determined.

• **Space problem**:

The terms of the corrective solution set $\Delta\mathcal{S}_{n+1}$ must be determined such as they verify the admissibility conditions of the original problem and at the same time verify the descent search direction. The verification of the admissibility condition is simple, it just suffices to consider the equilibrium equation at this global stage which is recalled below:

$\forall\underline{v} \in \mathcal{U}^S(\Omega,0)\otimes\mathcal{U}^T(I)$,

$$\int_{\Omega\times I}\rho\underline{\ddot{u}}_{n+1}\cdot\underline{v}\;d\Omega dt + \int_{\Omega\times I}\underline{\underline{\sigma}}_{n+1} : \underline{\underline{\varepsilon}}(\underline{v})\;d\Omega dt = \int_{\Omega\times I}\rho\underline{f}\cdot\underline{v}\;d\Omega dt + \int_{\partial_N\Omega\times I}\underline{f}^N\cdot\underline{v}\;dSdt \tag{4.48}$$

Now subtracting the equilibrium equations for both space solutions $\mathcal{S}_{n+1}$ and $\mathcal{S}_n$, leads to:

$\forall\underline{v} \in \mathcal{U}^S(\Omega,0)\otimes\mathcal{U}^T(I)$,

$$\int_{\Omega\times I}\rho\Delta\underline{\ddot{u}}_{n+1}\cdot\underline{v}\;d\Omega dt + \int_{\Omega\times I}\Delta\underline{\underline{\sigma}}_{n+1} : \underline{\underline{\varepsilon}}(\underline{v})\;d\Omega dt = 0 \tag{4.49}$$

The search direction (4.39) is now used in order to replace the stress correction and obtain an expression only in function of the displacement correction, obtaining:

$\forall\underline{v} \in \mathcal{U}^S(\Omega,0)\otimes\mathcal{U}^T(I)$,

$$\int_{\Omega\times I}\rho\Delta\underline{\ddot{u}}_{n+1}\cdot\underline{v}\;d\Omega dt + \int_{\Omega\times I}\underline{\underline{\varepsilon}}(\underline{v}) : \mathbb{H}_\varepsilon : \Delta\underline{\underline{\varepsilon}}_{n+1}\;d\Omega dt = \int_{\Omega\times I}\underline{\underline{\Delta}}_{n+1} : \underline{\underline{\varepsilon}}(\underline{v})\;d\Omega dt \tag{4.50}$$

Now lets use the low-rank approximation of equation (4.47) and introduce these into the equilibrium equation (4.50) to obtain:

$\forall\underline{\bar{v}} \in \mathcal{U}^S(\Omega,0)$,

$$\int_\Omega \rho < \ddot{\lambda}\lambda > \underline{\bar{u}}\cdot\underline{\bar{v}}\;d\Omega + \int_\Omega < \lambda\lambda > \underline{\underline{\bar{\varepsilon}}}(\underline{\bar{v}}) : \mathbb{H}_\varepsilon : \underline{\underline{\bar{\varepsilon}}}(\underline{\bar{u}})d\Omega = \int_\Omega < \underline{\underline{\Delta}}_{n+1}\lambda > : \underline{\underline{\bar{\varepsilon}}}(\underline{\bar{v}})\;d\Omega \tag{4.51}$$

where we denote $< \cdot >= \int_I (\cdot) dt$ to simplify the notations.

The above formulation only depends on the displacement term, therefore it can be easily solved by employing classical FEM. After calculating the corrective spatial displacement field $\bar{\underline{u}}(\underline{x}) \in \mathcal{U}^S(\Omega, 0)$, we compute the spatial corrective strain $\bar{\underline{\underline{\varepsilon}}}(\bar{\underline{u}})$ and finally, the space function of the corrective stress is computed by using the search direction (4.39):

$$\bar{\underline{\underline{\sigma}}}(\underline{x}) = \mathbb{H}_\varepsilon : \bar{\underline{\underline{\varepsilon}}}(\underline{x}) - \frac{\int_I \lambda \underline{\underline{\Delta}}_{n+1} dt}{\int_I \lambda^2 dt} \tag{4.52}$$

Once the space functions $\bar{\underline{u}}(\underline{x})$, $\bar{\underline{\underline{\varepsilon}}}(\underline{x})$ and $\bar{\underline{\underline{\sigma}}}(\underline{x})$ computed, the temporal function $\lambda(t)$ must be determined. This determination is shown below.

• **Temporal problem**:

As exposed in the introductory section, the temporal functions are determined such as the constitutive relation error of equation (4.41) is minimized, this is:

$$\{\lambda_{m+1}\} = \underset{\lambda_{m+1} \in \mathcal{U}^T}{\arg\min} \left| \left| \left| \bar{\underline{\underline{\sigma}}}_{m+1} \lambda_{m+1} - \mathbb{H}_\varepsilon : \bar{\underline{\underline{\varepsilon}}}_{m+1} \lambda_{m+1} + \underline{\underline{\Delta}}_{n+1} \right| \right| \right|^2_{\mathbb{H}_\varepsilon^{-1}} \tag{4.53}$$

By dropping out the index $m+1$ ($\lambda = \lambda_{m+1}$) and writing the stationarity condition of (4.53) with respect to λ, one reads:

$\forall \delta\lambda \in \mathcal{U}^T$,

$$\int_{\Omega \times I} \delta\lambda\lambda (\bar{\underline{\underline{\sigma}}} - \mathbb{H}_\varepsilon : \bar{\underline{\underline{\varepsilon}}}) : \mathbb{H}_\varepsilon^{-1} : (\bar{\underline{\underline{\sigma}}} - \mathbb{H}_\varepsilon : \bar{\underline{\underline{\varepsilon}}}) + \delta\lambda (\bar{\underline{\underline{\sigma}}} - \mathbb{H}_\varepsilon : \bar{\underline{\underline{\varepsilon}}}) : \mathbb{H}_\varepsilon^{-1} : \underline{\underline{\Delta}}_{n+1} \, d\Omega dt = 0 \tag{4.54}$$

Due to the constant choice of the operator $\mathbb{H}_\varepsilon$ we can define the following scalar value:

$$A^c = \int_\Omega (\bar{\underline{\underline{\sigma}}} - \mathbb{H}_\varepsilon : \bar{\underline{\underline{\varepsilon}}}) : \mathbb{H}_\varepsilon^{-1} : (\bar{\underline{\underline{\sigma}}} - \mathbb{H}_\varepsilon : \bar{\underline{\underline{\varepsilon}}}) \, d\Omega \tag{4.55}$$

and the temporal function:

$$D^c(t) = -\int_\Omega (\bar{\underline{\underline{\sigma}}} - \mathbb{H}_\varepsilon : \bar{\underline{\underline{\varepsilon}}}) : \mathbb{H}_\varepsilon^{-1} : \underline{\underline{\Delta}}_{n+1} \, d\Omega \tag{4.56}$$

which can be used in equation (4.54), to obtain the following compact formulation:

$\forall \delta\lambda \in \mathcal{U}^T$,

$$\int_I \delta\lambda\lambda A^c \, dt = \int_I \delta\lambda D^c(t) \, dt \tag{4.57}$$

As mentioned in the introduction, the temporal problem is solved by using the TDGM. The idea is to solve incrementally the function over the temporal domain using at the same time a finite element formulation. In this sense, the temporal FEM discretization of expression (4.57) applied at time element "k" is given as:

$$\underline{\underline{\boldsymbol{Q}}}^{[k]} \underline{\boldsymbol{\lambda}}^{[k]} = \underline{\boldsymbol{f}}^{[k]} \tag{4.58}$$

with:

$$\underline{\underline{\boldsymbol{Q}}}^{[k]} = \int_{\check{I}_k} \underline{\psi}^{[k]}(t) \otimes \underline{\psi}^{[k]}(t) A^c dt \quad , \quad \underline{\boldsymbol{f}}^{[k]} = \int_{\check{I}_k} \underline{\psi}^{[k]}(t) D^c(t) dt \tag{4.59}$$

and where $\underline{\boldsymbol{\lambda}}^{[k]}$ corresponds to the nodal values of the temporal function at time element k.

However, this equation is incomplete since it does not impose continuity between time intervals. To resolve this issue, continuity between time elements must be imposed in a weak sense as illustrated in figure 4.6.

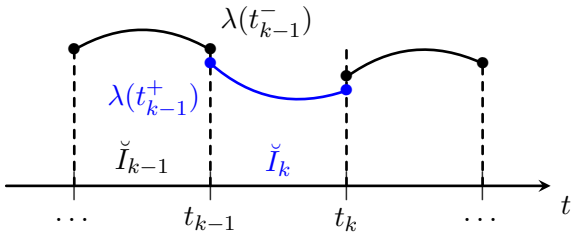


Figure 4.6: Weak imposition of the continuity between the intervals.

The idea consists in transmitting the end value of the time function of the interval $\breve{I}_{k-1}$ $(\lambda(t_{k-1}^-))$ to the initial value of the interval $\breve{I}_k$ $(\lambda(t_{k-1}^+))$, by using some operators that must be included in the discretized equation (4.58) (see figure 4.6). The easiest way to do this consists in defining the following operators:

$$\begin{aligned} \underline{\underline{\boldsymbol{\mathcal{L}}}}^{[k]} &= 1.1\,\max(\underline{\underline{\boldsymbol{Q}}}^{[k]})\underline{\psi}^{[k]}(t_{k-1}) \otimes \underline{\psi}^{[k]}(t_{k-1}^+) \\ \underline{\underline{\boldsymbol{\mathcal{R}}}}^{[k]} &= 1.1\,\max(\underline{\underline{\boldsymbol{Q}}}^{[k]})\underline{\psi}^{[k]}(t_{k-1}) \otimes \underline{\psi}^{[k-1]}(t_{k-1}^-) \end{aligned} \tag{4.60}$$

whose detailed representations are given by:

$$\underline{\underline{\boldsymbol{\mathcal{L}}}}^{[k]} = \begin{bmatrix} 1.1\,\max(\underline{\underline{\boldsymbol{Q}}}^{[k]}) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad , \quad \underline{\underline{\boldsymbol{\mathcal{R}}}}^{[k]} = \begin{bmatrix} 0 & 0 & 0 & 1.1\,\max(\underline{\underline{\boldsymbol{Q}}}^{[k]}) \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

These operators are introduced in the discretized equation of each time interval $\breve{I}_k$ in the following way:

$$\left(\underline{\underline{\boldsymbol{Q}}}^{[k]} + \underline{\underline{\boldsymbol{\mathcal{L}}}}^{[k]}\right) \underline{\boldsymbol{\lambda}}^{[k]} = \underline{\underline{\boldsymbol{\mathcal{R}}}}^{[k]}\, \underline{\boldsymbol{\lambda}}^{[k-1]} + \underline{\boldsymbol{f}}^{[k]} \tag{4.61}$$

The factor $1.1\,\max(\underline{\underline{\boldsymbol{Q}}}^{[k]})$, with $\max(\cdot)$ the function that finds the maximum value of the matrix $(\cdot)$, is chosen in order to add terms in the same order of magnitude compared to the discretized elemental matrix $\underline{\underline{\boldsymbol{Q}}}^{[k]}$ of the problem.

What these operators do simply consists in unbalancing the equation associated to the first nodal value of $\underline{\boldsymbol{\lambda}}^{[k]}$ (first row) of equation (4.61), such as the only way of approximately solving the original problem given in (4.58) is to have $\underline{\underline{\boldsymbol{\mathcal{L}}}}^{[k]}\, \underline{\boldsymbol{\lambda}}^{[k]} \approx \underline{\underline{\boldsymbol{\mathcal{R}}}}^{[k]}\, \underline{\boldsymbol{\lambda}}^{[k-1]}$, which is traduced in $\lambda(t_{k-1}^-) \approx \lambda(t_{k-1}^+)$.

Thanks to this formulation, the time functions are solved incrementally, imposing in a weak sense the continuity between each time interval of the discretization. This idea is also applied to the update of the temporal PGD functions at the preliminary step as shown in the next subsection.

4.3.2.2 Preliminary step

Since every time a new mode is calculated from the enrichment step, the question if the low-rank approximation can be re-arranged in order to better decrease the LATIN error without calculating additional PGD couples naturally arises. This can be achieved by simply actualizing the temporal functions of the PGD decomposition without modifying the spatial ones. This operation allows to accelerate the convergence to the solution of the problem while avoiding the computation of new space-time PGD pairs functions, allowing by consequence a most compressed solution representation. The details about the temporal update of PGD functions are given below.

4.3.2.2.1 Update of the temporal functions

The idea of the preliminary step consists in updating the temporal functions of all the global quantities while maintaining the space functions fixed, i.e:

$\forall i \in [1, ..., m+1]$,

$$\lambda_i \leftarrow \lambda_i + \Delta\lambda_i(t) \tag{4.62}$$

where the main unknowns are the time corrective functions $(\Delta\lambda_i(t))_{i=1}^{m+1}$. This procedure allows to actualize the PGD approximation of the global quantities as follows:

$$\begin{aligned}
\underline{u}_{n+1}(\underline{x}, t) &= \sum_{i=1}^{m+1} \underline{\bar{u}}_i(\underline{x}) \left[\lambda_i(t) + \Delta\lambda_i(t)\right] + \underline{u}_0(\underline{x}, t) \\
\underline{\underline{\varepsilon}}_{n+1}(\underline{x}, t) &= \sum_{i=1}^{m+1} \underline{\underline{\bar{\varepsilon}}}_i(\underline{x}) \left[\lambda_i(t) + \Delta\lambda_i(t)\right] + \underline{\underline{\varepsilon}}_0(\underline{x}, t) \\
\underline{\underline{\sigma}}_{n+1}(\underline{x}, t) &= \sum_{i=1}^{m+1} \underline{\underline{\bar{\sigma}}}_i(\underline{x}) \left[\lambda_i(t) + \Delta\lambda_i(t)\right] + \underline{\underline{\sigma}}_0(\underline{x}, t)
\end{aligned} \tag{4.63}$$

The time corrective functions are calculated by minimizing the constitutive relation error (4.41). In this sense, by introducing the expressions (4.63) into (4.41) we obtain:

$\forall i \in [1, ..., m+1]$,

$$\{\Delta\lambda_i\} = \underset{\{\Delta\lambda_i\}_{i=1}^{m+1} \in \mathcal{U}^T}{\arg\min} \left| \left| \left| \sum_{i=1}^{m+1} \underline{\underline{\bar{\sigma}}}_i \Delta\lambda_i - \mathbb{H}_\varepsilon : \left(\sum_{i=1}^{m+1} \underline{\underline{\bar{\varepsilon}}}_i \Delta\lambda_i \right) + \underline{\underline{\Delta}}_{n+1} \right| \right| \right|_{\mathbb{H}_\varepsilon^{-1}}^2 \tag{4.64}$$

As in the case of the enrichment step, the above problem can also be solved incrementally in time by applying the TDGM. For this, we first develop the minimization problem, to obtain:

$\forall \delta\Delta\lambda_i \in \mathcal{U}^T$, $\forall i \in [1, ..., m+1]$,

$$\int_{\Omega \times I} \left(\sum_{i=1}^{m+1} \left(\underline{\underline{\bar{\sigma}}}_i - \mathbb{H}_\varepsilon : \underline{\underline{\bar{\varepsilon}}}_i \right) \delta\Delta\lambda_i \right) : \mathbb{H}_\varepsilon^{-1} : \left(\sum_{j=1}^{m+1} \left(\underline{\underline{\bar{\sigma}}}_j - \mathbb{H}_\varepsilon : \underline{\underline{\bar{\varepsilon}}}_j \right) \Delta\lambda_j + \underline{\underline{\Delta}}_{n+1} \right) \, d\Omega dt = 0 \tag{4.65}$$

which can be simplified in:

$$\int_I \sum_{i=1}^{m+1} \sum_{j=1}^{m+1} \delta\Delta\lambda_i \Delta\lambda_j A_{ij}^c \, dt = \sum_{i=1}^{m+1} \delta\Delta\lambda_i D_i^c(t) \, dt \tag{4.66}$$

with the constants:

$\forall (i, j) \in [1, ..., m+1]$,

$$A_{ij}^c = \int_\Omega \left(\underline{\underline{\bar{\sigma}}}_i - \mathbb{H}_\varepsilon : \underline{\underline{\bar{\varepsilon}}}_i \right) : \mathbb{H}_\varepsilon^{-1} : \left(\underline{\underline{\bar{\sigma}}}_j - \mathbb{H}_\varepsilon : \underline{\underline{\bar{\varepsilon}}}_j \right) \, d\Omega \tag{4.67}$$

and the respective temporal functions:

$\forall i \in [1, ..., m+1]$,

$$D_i^c(t) = -\int_\Omega \left(\underline{\underline{\bar{\sigma}}}_i - \mathbb{H}_\varepsilon : \underline{\underline{\bar{\varepsilon}}}_i \right) : \mathbb{H}_\varepsilon^{-1} : \underline{\underline{\Delta}}_{n+1} \, d\Omega \tag{4.68}$$

As explained for the enrichment step, the discretization of expression (4.66) requires the definition of operators that allow to impose the continuity between the temporal intervals, but this time for all the $m+1$ functions considered, this is illustrated in figure 4.7.

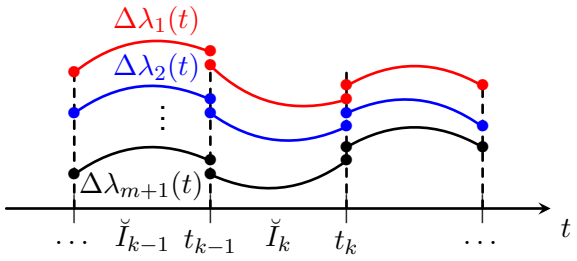


Figure 4.7: Weak imposition of the continuity between the intervals for all the temporal PGD functions.

In this sense, by applying the same ideas as previously explained for the enrichment step, the following discretized equation is obtained when applied to an element "k":

$\forall i \in [1, ..., m+1]$,

$$\sum_{j=1}^{m+1} \underline{\underline{\boldsymbol{Q}}}_{ij}^{[k]} \Delta\underline{\boldsymbol{\lambda}}_j^{[k]} + \sum_{j=1}^{m+1} \underline{\underline{\boldsymbol{\mathcal{L}}}}_{ij}^{[k]} \Delta\underline{\boldsymbol{\lambda}}_j^{[k]} = \sum_{j=1}^{m+1} \underline{\underline{\boldsymbol{\mathcal{R}}}}_{ij}^{[k]} \Delta\underline{\boldsymbol{\lambda}}_j^{[k-1]} + \underline{\boldsymbol{f}}_i^{[k]} \tag{4.69}$$

The above expression can be condensed into:

$$\left(\underline{\underline{\boldsymbol{Q}}}_{up}^{[k]} + \underline{\underline{\boldsymbol{\mathcal{L}}}}_{up}^{[k]} \right) \Delta\underline{\boldsymbol{\Lambda}}^{[k]} = \underline{\underline{\boldsymbol{\mathcal{R}}}}_{up}^{[k]} \Delta\underline{\boldsymbol{\Lambda}}^{[k-1]} + \underline{\boldsymbol{f}}_{up}^{[k]} \tag{4.70}$$

with the vector containing the nodal values of all the temporal functions given by:

$$\Delta\underline{\boldsymbol{\Lambda}} = \begin{bmatrix} \Delta\underline{\boldsymbol{\lambda}}_1 \\ \Delta\underline{\boldsymbol{\lambda}}_2 \\ \vdots \\ \Delta\underline{\boldsymbol{\lambda}}_{m+1} \end{bmatrix} \tag{4.71}$$

The full expression of the matrices are given by:

$$\underline{\underline{\boldsymbol{Q}}}_{up}^{[k]} = \begin{bmatrix} \underline{\underline{\boldsymbol{Q}}}_{11}^{[k]} & \underline{\underline{\boldsymbol{Q}}}_{12}^{[k]} & \cdots & \underline{\underline{\boldsymbol{Q}}}_{1\ m+1}^{[k]} \\ \underline{\underline{\boldsymbol{Q}}}_{21}^{[k]} & \ddots & & \\ \vdots & & & \\ \underline{\underline{\boldsymbol{Q}}}_{m+1\ 1}^{[k]} & & & \underline{\underline{\boldsymbol{Q}}}_{m+1\ m+1}^{[k]} \end{bmatrix} \quad , \quad \underline{\boldsymbol{f}}_{up}^{[k]} = \begin{bmatrix} \underline{\boldsymbol{f}}_1^{[k]} \\ \underline{\boldsymbol{f}}_2^{[k]} \\ \vdots \\ \underline{\boldsymbol{f}}_{m+1}^{[k]} \end{bmatrix}$$

with:

$\forall (i,j) \in [1, ..., m+1]$,

$$\begin{aligned} \underline{\underline{\boldsymbol{Q}}}_{ij}^{[k]} &= \int_{\breve{I}_k} \underline{\psi}^{[k]}(t) \otimes \underline{\psi}^{[k]}(t) A_{ij}^c dt \\ \underline{\boldsymbol{f}}_i^{[k]} &= \int_{\breve{I}_k} \underline{\psi}^{[k]}(t) D_i^c(t) dt \end{aligned} \tag{4.72}$$

and the matrices that allow the weak imposition of the initial conditions over all the temporal functions:

$$\underline{\underline{\boldsymbol{\mathcal{L}}}}^{[k]}_{up} = \begin{bmatrix} \underline{\underline{\boldsymbol{\mathcal{L}}}}^{[k]}_{11} & \underline{\underline{\boldsymbol{\mathcal{L}}}}^{[k]}_{12} & \cdots & \underline{\underline{\boldsymbol{\mathcal{L}}}}^{[k]}_{1\ m+1} \\ \underline{\underline{\boldsymbol{\mathcal{L}}}}^{[k]}_{21} & \ddots & & \\ \vdots & & & \\ \underline{\underline{\boldsymbol{\mathcal{L}}}}^{[k]}_{m+1\ 1} & & & \underline{\underline{\boldsymbol{\mathcal{L}}}}^{[k]}_{m+1\ m+1} \end{bmatrix} \quad , \quad \underline{\underline{\boldsymbol{\mathcal{R}}}}^{[k]}_{up} = \begin{bmatrix} \underline{\underline{\boldsymbol{\mathcal{R}}}}^{[k]}_{11} & \underline{\underline{\boldsymbol{\mathcal{R}}}}^{[k]}_{12} & \cdots & \underline{\underline{\boldsymbol{\mathcal{R}}}}^{[k]}_{1\ m+1} \\ \underline{\underline{\boldsymbol{\mathcal{R}}}}^{[k]}_{21} & \ddots & & \\ \vdots & & & \\ \underline{\underline{\boldsymbol{\mathcal{R}}}}^{[k]}_{m+1\ 1} & & & \underline{\underline{\boldsymbol{\mathcal{R}}}}^{[k]}_{m+1\ m+1} \end{bmatrix}$$

with:

$\forall (i,j) \in [1, ..., m+1]\,,$

$$\underline{\underline{\boldsymbol{\mathcal{L}}}}^{[k]}_{ij} = 1.1 \max(\underline{\underline{\boldsymbol{Q}}}^{[k]}_{ij}) \underline{\psi}^{[k]}(t_{k-1}) \otimes \underline{\psi}^{[k]}(t^{+}_{k-1}) \tag{4.73}$$

$$\underline{\underline{\boldsymbol{\mathcal{R}}}}^{[k]}_{ij} = 1.1 \max(\underline{\underline{\boldsymbol{Q}}}^{[k]}_{ij}) \underline{\psi}^{[k]}(t_{k-1}) \otimes \underline{\psi}^{[k-1]}(t^{-}_{k-1}) \tag{4.74}$$

The discretized equation (4.70) allows the imposition of the continuity between the intervals $\breve{I}_k$ and $\breve{I}_{k-1}$ for all the involved corrective temporal PGD functions. Therefore, its resolution can be done incrementally over the whole time domain.

To control the iterative improvement of the temporal PGD basis, every time the actualization of the temporal functions is done, an indicator of stagnation of the updated functions is calculated as follows:

$$\zeta_{up} = \max \left(\left\| 0.5 \left(\frac{|\Delta \underline{\boldsymbol{\Lambda}}|}{|\underline{\boldsymbol{\Lambda}}| + |\Delta \underline{\boldsymbol{\Lambda}}|} \right) \right\|_2 \right) \tag{4.75}$$

where $\|\cdot\|_2$ denotes the euclidean norm, and the terms $\underline{\boldsymbol{\Lambda}}$ and $\Delta \underline{\boldsymbol{\Lambda}}$ correspond to the vectors that contain all the time discretized PGD functions and their respective corrections computed at the current preliminary step.

Remark: After each preliminary step, the local stage must be evaluated as presented in the diagram of figure 4.2. Once the update indicator reaches a certain threshold, the preliminary step is stopped and a new PGD mode is calculated in the enrichment step, starting a preliminary step process again.

4.3.3 LATIN error indicator

Once the global and local stages are solved, an error indicator is computed in order to assess the quality of the solution obtained. The error indicator can be defined in different ways, however, herein this error is defined with the objective of being fast to evaluate and give a real distance between the local and global quantities of interest. This error is simply defined as follows:

$$\xi_{n+1} = 100 \sqrt{\left(\frac{\left| \left| \left| \underline{\underline{\sigma}}_{n+1} - \hat{\underline{\underline{\sigma}}}_{n+1/2} \right| \right| \right|^2}{\left| \left| \left| \underline{\underline{\sigma}}_{n+1} \right| \right| \right|^2} + \frac{\left| \left| \left| \underline{\underline{\varepsilon}}_{n+1} - \hat{\underline{\underline{\varepsilon}}}_{n+1/2} \right| \right| \right|^2}{\left| \left| \left| \underline{\underline{\varepsilon}}_{n+1} \right| \right| \right|^2} \right) [\%]} \tag{4.76}$$

with the used norm given by:

$$\left| \left| \left| \underline{\underline{\bullet}} \right| \right| \right|^2 = \int_{\Omega \times I} (\underline{\underline{\bullet}}) : (\underline{\underline{\bullet}})\, d\Omega dt$$

The LATIN-PGD method is said to converge when the above indicator is less than a given threshold.

4.4 Elasto-visco-plastic material

The present section details the application of the LATIN-PGD strategy for solving nonlinear dynamic problems involving this time elasto-visco-plastic constitutive relations. Dealing with a constitutive law of different nature is not straightforward. Compared to what has been presented in section 4.3, of course, the local stage, where constitutive relations are solved, has to be modified but also the search directions must be changed and the global stage itself must be deeply adapted. For completeness, every sub-steps are detailed, once again, for this new kind of material nonlinearity.

As was done in previous section 4.3, the following subsections detail the iterative resolution process between the local and global stages by assuming that a global solution $\mathcal{S}_n \in \boldsymbol{A_d}$ is known.

4.4.1 Local nonlinear stage

After the resolution of the global stage $\mathcal{S}_n \in \boldsymbol{A_d}$ where the admissibility conditions are verified, the constitutive relation quantities $\hat{\mathcal{S}}_{n+1/2} \in \boldsymbol{\Gamma}$ are computed at the local stage. For the case of elasto-visco-plasticity

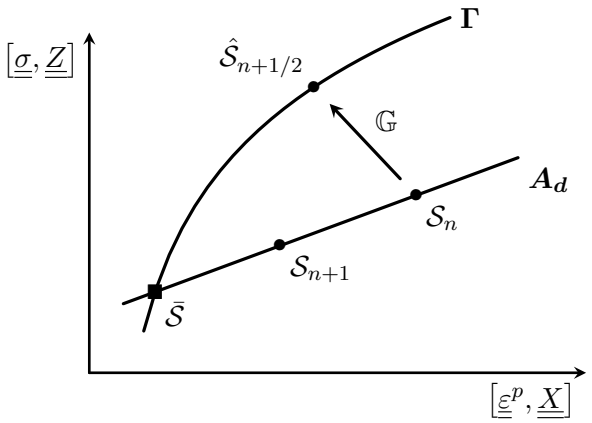


Figure 4.8: Calculation of the local solution of space $\boldsymbol{\Gamma}$.

the local quantities that must be determined are:

$$\hat{\mathcal{S}}_{n+1/2} = \{\hat{\underline{\underline{\sigma}}}_{n+1/2}, \hat{\underline{\underline{\varepsilon}}}^p_{n+1/2}, \hat{\underline{\underline{\beta}}}_{n+1/2}, \hat{\underline{\underline{\alpha}}}_{n+1/2}, \hat{\bar{r}}_{n+1/2}, \hat{\bar{R}}_{n+1/2}\} \tag{4.77}$$

where $\underline{\underline{\alpha}}$ and $\bar{r}$ corresponds to the kinematic and isotropic hardening internal variables with $\underline{\underline{\beta}}$ and $\bar{R}$ their corresponding dual representation; $\underline{\underline{\varepsilon}}^p$ corresponds to the plastic deformation.

The local solution must also verify the ascent search direction, this is:

$$\boxed{\text{Find} \quad \hat{\mathcal{S}}_{n+1/2} \in \boldsymbol{\Gamma} \quad \text{such that} \quad \left(\hat{\mathcal{S}}_{n+1/2} - \mathcal{S}_n\right) \in \mathbb{G}} \tag{4.78}$$

where the global quantities at iteration n of the LATIN method are given by:

$$\mathcal{S}_n = \{\underline{u}_n, \underline{\underline{\varepsilon}}_n, \underline{\underline{\sigma}}_n, \underline{\underline{\varepsilon}}^p_n, \underline{\underline{\beta}}_n, \underline{\underline{\alpha}}_n, \bar{r}_n, \bar{R}_n\} \tag{4.79}$$

The idea of equation (4.78) is illustrated in figure 4.8, in where the following definition is used for the internal variables (see chapter 2):

$$\underline{\underline{\dot{X}}} = \left[\underline{\underline{\dot{\alpha}}}, \dot{\bar{r}}\right]^T \qquad \text{and} \qquad \underline{\underline{Z}} = \left[\underline{\underline{\beta}}, \bar{R}\right]^T \tag{4.80}$$

The ascent search direction written in mathematical form is given by:

$$\mathbb{G}: \quad \left\{ \begin{bmatrix} \hat{\dot{\underline{\underline{\varepsilon}}}}^p_{n+1/2} - \dot{\underline{\underline{\varepsilon}}}^p_n \\ -(\hat{\dot{\underline{\underline{X}}}}_{n+1/2} - \dot{\underline{\underline{X}}}_n) \end{bmatrix} + \hat{\mathbb{H}}: \begin{bmatrix} \hat{\underline{\underline{\sigma}}}_{n+1/2} - \underline{\underline{\sigma}}_n \\ \hat{\underline{\underline{Z}}}_{n+1/2} - \underline{\underline{Z}}_n \end{bmatrix} = 0 \right. \tag{4.81}$$

where $\hat{\mathbb{H}}$ is any positive definite operator. As presented for the case of isotropic damage, here we also choose a constant ascent search direction, meaning that:

$$\hat{\mathbb{H}}^{-1} = 0 \tag{4.82}$$

This condition allows to easily determine the local quantities related to the stress and the internal variables as:

$$\begin{aligned} \hat{\underline{\underline{\sigma}}}_{n+1/2} &= \underline{\underline{\sigma}}_n \\ \hat{\underline{\underline{Z}}}_{n+1/2} &= \underline{\underline{Z}}_n = \left[\underline{\underline{\beta}}_n, \bar{R}_n \right] \end{aligned} \tag{4.83}$$

The remaining quantities related to the rate of plastic deformation and internal variables are simply determined by verifying the elasto-visco-plasticity constitutive relation, which is recalled below under a synthetic form:

$$\begin{bmatrix} \dot{\underline{\underline{\varepsilon}}}^p(\underline{x},t) \\ \dot{\underline{\underline{X}}}(\underline{x},t) \end{bmatrix} = B\left(\underline{\underline{\sigma}}(\underline{x},t), \underline{\underline{Z}}(\underline{x},t) \right)$$

where $\underline{\underline{\varepsilon}}^p$ stands for the plastic deformation and $\underline{\underline{X}}$ the internal variables. By using the full expression of the above equation (see chapter 2), the remaining local quantities are determined as:

$$\begin{aligned} \hat{\dot{\underline{\underline{\varepsilon}}}}^p_{n+1/2}(\underline{x},t) &= k\langle f_s \rangle_+^{n_s} \left[\frac{3}{2} \frac{\hat{\underline{\underline{\tau}}}_{n+1/2}}{\sqrt{\frac{3}{2}\hat{\underline{\underline{\tau}}}_{n+1/2} : \hat{\underline{\underline{\tau}}}_{n+1/2}}} \right] \\ \hat{\dot{\underline{\underline{\alpha}}}}_{n+1/2}(\underline{x},t) &= -k\langle f_s \rangle_+^{n_s} \left[-\frac{3}{2} \frac{\hat{\underline{\underline{\tau}}}_{n+1/2}}{\sqrt{\frac{3}{2}\hat{\underline{\underline{\tau}}}_{n+1/2} : \hat{\underline{\underline{\tau}}}_{n+1/2}}} + \frac{a}{C} \hat{\underline{\underline{\beta}}}_{n+1/2} \right] \\ \hat{\dot{r}}_{n+1/2}(\underline{x},t) &= -k\langle f_s \rangle_+^{n_s} \left(\frac{\hat{\bar{R}}_{n+1/2}}{R_\infty} \frac{b}{2} - b^{\frac{1}{2}} \right) \end{aligned} \tag{4.84}$$

with:

$$\hat{\underline{\underline{\tau}}}_{n+1/2} = \hat{\underline{\underline{\sigma}}}^D_{n+1/2} - \hat{\underline{\underline{\beta}}}_{n+1/2} \tag{4.85}$$

The variables $\hat{\underline{\underline{\sigma}}}^D_{n+1/2}$, $\hat{\underline{\underline{\alpha}}}_{n+1/2}$ and $\hat{\bar{r}}_{n+1/2}$ correspond respectively to the deviatoric stress tensor, the kinematic and the isotropic hardening respectively, with the plasticity threshold function given by:

$$f_s = \sqrt{\frac{3}{2}\hat{\underline{\underline{\tau}}}_{n+1/2} : \hat{\underline{\underline{\tau}}}_{n+1/2}} + \frac{a}{2C} \hat{\underline{\underline{\beta}}}_{n+1/2} : \hat{\underline{\underline{\beta}}}_{n+1/2} - R_\infty \left(\frac{\hat{\bar{R}}_{n+1/2}}{R_\infty} \frac{b^{\frac{1}{2}}}{2} \right) \left(2 - \frac{\hat{\bar{R}}_{n+1/2}}{R_\infty} \frac{b^{\frac{1}{2}}}{2} \right) - \sigma_y \tag{4.86}$$

Remark: The local stage quantities for the case of elasto-visco-plasticity are locally determined over the space and time domain, which allows to introduce parallel strategies for its resolution.

4.4.2 Global linear stage: equilibrium and compatibility equations

Once the local nonlinear solution $\hat{\mathcal{S}}_{n+1/2}$ is calculated, we compute the solution $\mathcal{S}_{n+1}$ on the linear space $\boldsymbol{A_d}$. In the global stage, apart from the state laws and admissibility conditions, the solution set $\mathcal{S}_{n+1} \in \boldsymbol{A_d}$ must also satisfy the descent search direction:

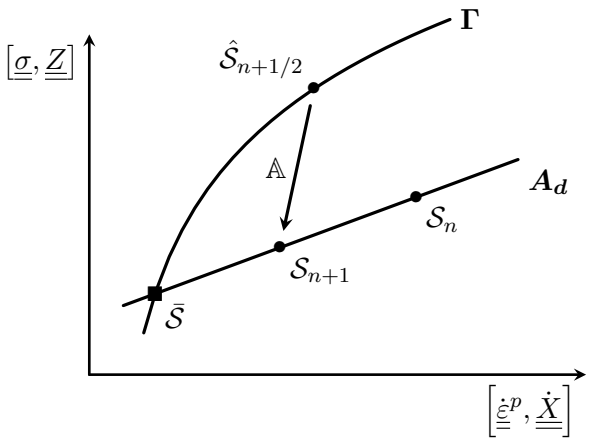


Figure 4.9: Calculation of the global solution $\mathcal{S}_{n+1} \in \boldsymbol{A_d}$.

This requirement can be stated as follows:

$$\boxed{\text{Find} \quad \mathcal{S}_{n+1} \in \boldsymbol{A_d} \quad \text{such that} \quad \left(\mathcal{S}_{n+1} - \hat{\mathcal{S}}_{n+1/2}\right) \in \mathbb{A}} \tag{4.87}$$

where the global solution set at LATIN iteration $n+1$ is given by:

$$\mathcal{S}_{n+1} = \{\underline{u}_{n+1}, \underline{\underline{\varepsilon}}_{n+1}, \underline{\underline{\varepsilon}}^p_{n+1}, \underline{\underline{\sigma}}_{n+1}, \underline{\underline{\beta}}_{n+1}, \underline{\underline{\alpha}}_{n+1}, \bar{r}_{n+1}, \bar{R}_{n+1}\} \tag{4.88}$$

The requirement of equation (4.87) can be mathematically written as the following descent search direction:

$$\mathbb{A}: \quad \left\{ \begin{bmatrix} \dot{\underline{\underline{\varepsilon}}}^p_{n+1} - \dot{\hat{\underline{\underline{\varepsilon}}}}^p_{n+1/2} \\ -(\dot{\underline{\underline{X}}}_{n+1} - \dot{\hat{\underline{\underline{X}}}}_{n+1/2}) \end{bmatrix} \right\} - \mathbb{H}: \begin{bmatrix} \underline{\underline{\sigma}}_{n+1} - \hat{\underline{\underline{\sigma}}}_{n+1/2} \\ \underline{\underline{Z}}_{n+1} - \hat{\underline{\underline{Z}}}_{n+1/2} \end{bmatrix} = 0 \tag{4.89}$$

where:

$$\dot{\underline{\underline{X}}}_{n+1} = \begin{bmatrix} \dot{\underline{\underline{\alpha}}}_{n+1} \\ \dot{\bar{r}}_{n+1} \end{bmatrix} \qquad \text{and} \qquad \underline{\underline{Z}}_{n+1} = \begin{bmatrix} \underline{\underline{\beta}}_{n+1} \\ \bar{R}_{n+1} \end{bmatrix} \tag{4.90}$$

and with the operator $\mathbb{H}$ given by:

$$\mathbb{H} = \begin{bmatrix} \mathbb{H}_\sigma & 0 \\ 0 & \mathbb{H}_Z \end{bmatrix} \quad , \quad \text{with} \quad \mathbb{H}_Z = \begin{bmatrix} \mathbb{H}_\beta & 0 \\ 0 & H_{\bar{R}} \end{bmatrix} \tag{4.91}$$

As for the isotropic damage case, the operators $\mathbb{H}_\sigma$, $\mathbb{H}_\beta$ and $H_{\bar{R}}$ are chosen to be constant to reduce the complexity of the LATIN-PGD implementation that will be presented in following sections, in this sense, the operators are chosen as follows:

$$\mathbb{H}_\sigma = c(\mathbb{K})^{-1} \tag{4.92}$$

$$\mathbb{H}_\beta = (\mathbb{K})^{-1} \tag{4.93}$$

$$H_{\bar{R}} = R_\infty^{-1} \tag{4.94}$$

with the factor $c = 10^{-4}$. Constants operators allows a faster global stage resolution, however, this is just a choice and the implementation of a tangent operator could also be considered without restrictions. The operator $\mathbb{H}_\sigma$ is constructed in a different way as seen from equation (4.92), which involves the use of a factor c, the explanation about this construction is left to the enrichment step section, where an analysis must be carried out for its determination in order to improve the LATIN-PGD convergence properties.

The expression of the descent search direction of equation (4.89) can be fully written as:

$$\mathbb{A}: \quad \begin{cases} \dot{\underline{\underline{\varepsilon}}}^p_{n+1} - \hat{\dot{\underline{\underline{\varepsilon}}}}^p_{n+1/2} - \mathbb{H}_\sigma : \left(\underline{\underline{\sigma}}_{n+1} - \hat{\underline{\underline{\sigma}}}_{n+1/2}\right) = 0 \\ -(\dot{\underline{\underline{\alpha}}}_{n+1} - \hat{\dot{\underline{\underline{\alpha}}}}_{n+1/2}) - \mathbb{H}_\beta : (C\,\underline{\underline{\alpha}}_{n+1} - \hat{\underline{\underline{\beta}}}_{n+1/2}) = 0 \\ -(\dot{\bar{r}}_{n+1} - \hat{\dot{\bar{r}}}_{n+1/2}) - H_{\bar{R}}(R_\infty\,\bar{r}_{n+1} - \hat{\bar{R}}_{n+1/2}) = 0 \end{cases} \tag{4.95}$$

where in the above expression the following relation is used (see section 2.3.2):

$$\underline{Z}_{n+1} = \underline{\bar{\Upsilon}}\,\underline{X}_{n+1} \quad , \quad \underline{\bar{\Upsilon}} = \begin{bmatrix} C & 0 \\ 0 & R_\infty \end{bmatrix}$$

In the same way as done for the case of isotropic damage, in order to solve the global quantities at iteration $n+1$ of the LATIN method, a corrective solution is computed, such as:

$$\mathcal{S}_{n+1} = \Delta\mathcal{S}_{n+1} + \mathcal{S}_n \tag{4.96}$$

with the corrective term solution given by:

$$\Delta\mathcal{S}_{n+1} = \{\Delta\underline{u}_{n+1}, \Delta\underline{\underline{\varepsilon}}_{n+1}, \Delta\underline{\underline{\varepsilon}}^p_{n+1}, \Delta\underline{\underline{\sigma}}_{n+1}, \Delta\underline{\underline{\beta}}_{n+1}, \Delta\underline{\underline{\alpha}}_{n+1}, \Delta\bar{r}_{n+1}, \Delta\bar{R}_{n+1}\} \tag{4.97}$$

By rewriting the expressions of equation (4.95) by considering the corrective terms that are the main unknowns of the global stage and using the result of (4.83), we have:

$$\Delta\dot{\underline{\underline{\varepsilon}}}^p_{n+1} - \mathbb{H}_\sigma : \Delta\underline{\underline{\sigma}}_{n+1} + \underline{\underline{\Delta}}_{n+1} = 0 \tag{4.98}$$

$$\Delta\dot{\underline{\underline{\alpha}}}_{n+1} + \mathbb{H}_\beta : \left(C\Delta\underline{\underline{\alpha}}_{n+1}\right) + \underline{\underline{\Delta}}^\alpha_{n+1} = 0 \tag{4.99}$$

$$\Delta\dot{\bar{r}}_{n+1} + H_{\bar{R}}R_\infty\Delta\bar{r}_{n+1} + \Delta^{\bar{r}}_{n+1} = 0 \tag{4.100}$$

with the residual terms given by:

$$\underline{\underline{\Delta}}_{n+1} = \dot{\underline{\underline{\varepsilon}}}^p_n - \hat{\dot{\underline{\underline{\varepsilon}}}}^p_{n+1/2} \tag{4.101}$$

$$\underline{\underline{\Delta}}^\alpha_{n+1} = \dot{\underline{\underline{\alpha}}}_n - \hat{\dot{\underline{\underline{\alpha}}}}_{n+1/2} \tag{4.102}$$

$$\Delta^{\bar{r}}_{n+1} = \dot{\bar{r}}_n - \hat{\dot{\bar{r}}}_{n+1/2} \tag{4.103}$$

In the same way as presented in section 4.3, the above expressions lead to the definition of the constitutive relation errors (CRE) related to the different search directions and defined as follows:

$$J^p = \left|\left|\left|\Delta\dot{\underline{\underline{\varepsilon}}}^p_{n+1} - \mathbb{H}_\sigma : \Delta\underline{\underline{\sigma}}_{n+1} + \underline{\underline{\Delta}}_{n+1}\right|\right|\right|^2_{\mathbb{H}^{-1}_\sigma} \tag{4.104}$$

$$J^\alpha = \left|\left|\left|\Delta\dot{\underline{\underline{\alpha}}}_{n+1} + \mathbb{H}_\beta : \left(C\Delta\underline{\underline{\alpha}}_{n+1}\right) + \underline{\underline{\Delta}}^\alpha_{n+1}\right|\right|\right|^2_{\mathbb{H}^{-1}_\beta} \tag{4.105}$$

$$J^{\bar{r}} = \left|\left|\left|\Delta\dot{\bar{r}}_{n+1} + H_{\bar{R}}R_\infty\Delta\bar{r}_{n+1} + \Delta^{\bar{r}}_{n+1}\right|\right|\right|^2_{H^{-1}_{\bar{R}}} \tag{4.106}$$

with the norms given by:

$$
\begin{aligned}
|||\cdot|||^2_{\mathbb{H}_\sigma^{-1}} &= \int_{\Omega\times I} (\cdot) : \mathbb{H}_\sigma^{-1} : (\cdot)\, d\Omega dt \\
|||\cdot|||^2_{\mathbb{H}_\beta^{-1}} &= \int_{\Omega\times I} (\cdot) : \mathbb{H}_\beta^{-1} : (\cdot)\, d\Omega dt \\
|||\cdot|||^2_{H_{\bar{R}}^{-1}} &= \int_{\Omega\times I} (\cdot)\, H_{\bar{R}}^{-1}\, (\cdot)\, d\Omega dt
\end{aligned}
\tag{4.107}
$$

Therefore, in order to decrease the distance between the global and local quantities, i.e, achieving convergence of the LATIN solver, the corrective terms must be determined such as minimizing the CRE J^p, J^α and $J^{\bar{r}}$ [Passieux, 2008].

For the fast resolution of the global solution $\mathcal{S}_{n+1}$, the PGD is used. In this sense, we consider that the global stage solution at iteration n of the LATIN method is simply approximated by m PGD terms as follows:

$$
\begin{aligned}
\underline{u}_n(\underline{x},t) &= \sum_{i=1}^{m} \underline{\bar{u}}_i(\underline{x})\lambda_i(t) + \underline{u}_0(\underline{x},t) & \qquad \underline{\dot{\varepsilon}}^p_n(\underline{x},t) &= \sum_{i=1}^{m} \underline{\bar{\varepsilon}}^p_i(\underline{x})\dot{\lambda}_i(t) \\
\underline{\underline{\varepsilon}}_n(\underline{x},t) &= \sum_{i=1}^{m} \underline{\underline{\bar{\varepsilon}}}_i(\underline{x})\lambda_i(t) + \underline{\underline{\varepsilon}}_0(\underline{x},t) \quad , & \underline{\underline{\dot{\alpha}}}_n(\underline{x},t) &= \sum_{i=1}^{m} \underline{\underline{\bar{\alpha}}}_i(\underline{x})\dot{\lambda}^\alpha_i(t) \\
\underline{\underline{\sigma}}_n(\underline{x},t) &= \sum_{i=1}^{m} \underline{\underline{\bar{\sigma}}}_i(\underline{x})\lambda_i(t) + \underline{\underline{\sigma}}_0(\underline{x},t) & \qquad \dot{\bar{r}}_n(\underline{x},t) &= \sum_{i=1}^{m} \bar{\bar{r}}_i(\underline{x})\dot{\lambda}^{\bar{r}}_i(t)
\end{aligned}
\tag{4.108}
$$

The corrective solution set $\Delta\mathcal{S}_{n+1}$, like the case of isotropic damage material, is constructed on two principal steps, the enrichment and the preliminary steps. Despite the similarity of the steps to be solved compared to the isotropic damage case, their resolution differ, since the admissibility equations and the minimization of the constitutive relations errors depend on the material considered. These differences are presented in details in the coming sections along with the full presentation of these steps.

Remark: After the determination of the corrective solution set $\Delta\mathcal{S}_{n+1}$, a relaxation of the solution is applied in order to ensure the convergence of the method:

$$
\mathcal{S}_{n+1} \leftarrow (1-\mu)\mathcal{S}_n + \mu\mathcal{S}_{n+1} \tag{4.109}
$$

where the relaxation coefficient for the case of elasto-visco-plasticity is chosen equal to $\mu = 0.4$ [Ladevèze, 1999].

4.4.2.1 Enrichment step: Enrichment of the PGD approximation

As exposed in previous sections, on the enrichment step the low-rank approximation of the global stage solution is enriched. This is simply achieved by approximating the corrective global stage quantities as a rank one decomposition, this is:

$$
\begin{aligned}
\Delta\underline{u}_{n+1}(\underline{x},t) &= \underline{\bar{u}}_{m+1}(\underline{x})\lambda_{m+1}(t) \\
\Delta\underline{\underline{\varepsilon}}_{n+1}(\underline{x},t) &= \underline{\underline{\bar{\varepsilon}}}_{m+1}(\underline{x})\lambda_{m+1}(t) \\
\Delta\underline{\underline{\sigma}}_{n+1}(\underline{x},t) &= \underline{\underline{\bar{\sigma}}}_{m+1}(\underline{x})\lambda_{m+1}(t) \quad , \quad & \Delta\underline{\underline{\dot{\alpha}}}_{n+1}(\underline{x},t) &= \underline{\underline{\bar{\alpha}}}_{m+1}(\underline{x})\dot{\lambda}^\alpha_{m+1}(t) \\
\Delta\underline{\dot{\varepsilon}}^p_{n+1}(\underline{x},t) &= \underline{\bar{\varepsilon}}^p_{m+1}(\underline{x})\dot{\lambda}_{m+1}(t) & \Delta\dot{\bar{r}}_{n+1}(\underline{x},t) &= \bar{\bar{r}}_{m+1}(\underline{x})\dot{\lambda}^{\bar{r}}_{m+1}(t)
\end{aligned}
\tag{4.110}
$$

The above corrective terms are separated into two groups, those that need to verify the admissibility conditions, which are the displacement, strain, stress and plastic deformation ($\Delta\underline{u}_{n+1}$, $\Delta\underline{\underline{\varepsilon}}_{n+1}$, $\Delta\underline{\underline{\sigma}}_{n+1}$ and $\Delta\dot{\underline{\underline{\varepsilon}}}^p_{n+1}$), and the internal variables terms ($\Delta\dot{\underline{\underline{\alpha}}}_{n+1}$ and $\Delta\dot{\bar{r}}_{n+1}$).

The spatial and temporal functions are calculated in an iterative way by using a fixed point strategy. After each iteration when a space-time couple of functions is calculated a stagnation error is defined in order to stop the iterations. This procedure is applied to the admissibility terms and the internal variables whose details are presented in the following sections for each respective case.

4.4.2.1.1 Space and time resolution associated to the admissibility quantities

The enrichment stage related to the admissibility quantities is solved in an alternate fixed point strategy. To control the iterations we define a stagnation measure. This indicator will be given in function of the temporal function, similar as exposed in section 4.3; to do so, we normalize the spatial functions by the following constant:

$$c_p = \left\|\bar{\underline{\underline{\varepsilon}}}^p\right\|_\Omega \quad , \quad \|\cdot\|_\Omega = \sqrt{\int_\Omega (\cdot) : (\cdot)\, d\Omega} \tag{4.111}$$

where the spatial and temporal functions are scaled as follows:

$$\begin{array}{l} \bar{\underline{u}}_{m+1} \leftarrow \bar{\underline{u}}_{m+1}/c_p \\ \bar{\underline{\underline{\varepsilon}}}_{m+1} \leftarrow \bar{\underline{\underline{\varepsilon}}}_{m+1}/c_p \\ \bar{\underline{\underline{\sigma}}}_{m+1} \leftarrow \bar{\underline{\underline{\sigma}}}_{m+1}/c_p \\ \bar{\underline{\underline{\varepsilon}}}^p_{m+1} \leftarrow \bar{\underline{\underline{\varepsilon}}}^p_{m+1}/c_p \end{array} \quad , \quad \begin{array}{l} \lambda_{m+1} \leftarrow \lambda_{m+1}\, c_p \\ \dot{\lambda}_{m+1} \leftarrow \dot{\lambda}_{m+1}\, c_p \\ \ddot{\lambda}_{m+1} \leftarrow \ddot{\lambda}_{m+1}\, c_p \end{array}$$

The stagnation indicator of the fixed point process is therefore defined as:

$$\zeta_{PGD} = \frac{\left\| |\lambda^{(i)}_{m+1}| - |\lambda^{(i-1)}_{m+1}| \right\|_I}{\left\| |\lambda^{(i)}_{m+1}| + |\lambda^{(i-1)}_{m+1}| \right\|_I} \quad , \quad \|\cdot\|_I = \sqrt{\int_I (\cdot)^2 dt} \tag{4.112}$$

where $\lambda^{(i)}_{m+1}$ and $\lambda^{(i-1)}_{m+1}$ stands for the temporal mode at iteration i and $i-1$ respectively.

The strategy is stopped when the indicator is below a given threshold or when a maximum number of iterations is reached. At the end of the process the spatial and temporal functions are kept and added to the current low-rank approximation.

The space and time functions of the PGD decomposition are determined using a mixed formulation following the same idea as for the isotropic damage case. On one hand, the space functions are solved by a Galerkin projection on the equilibrium equations and on the other hand, the temporal functions are obtained by minimizing the constitutive relation error.

In what follows, for the case of the spatial and temporal problem, since the new mode to be calculated corresponds to the $m+1$, the indexes of the PGD mode are dropped out in order to alleviate the notations:

$$\begin{array}{l} \Delta\underline{u}_{n+1}(\underline{x},t) = \bar{\underline{u}}_{m+1}(\underline{x})\lambda_{m+1}(t) = \bar{\underline{u}}(\underline{x})\lambda(t) \\ \Delta\underline{\underline{\varepsilon}}_{n+1}(\underline{x},t) = \bar{\underline{\underline{\varepsilon}}}_{m+1}(\underline{x})\lambda_{m+1}(t) = \bar{\underline{\underline{\varepsilon}}}(\underline{x})\lambda(t) \\ \Delta\underline{\underline{\sigma}}_{n+1}(\underline{x},t) = \bar{\underline{\underline{\sigma}}}_{m+1}(\underline{x})\lambda_{m+1}(t) = \bar{\underline{\underline{\sigma}}}(\underline{x})\lambda(t) \\ \Delta\dot{\underline{\underline{\varepsilon}}}^p_{n+1}(\underline{x},t) = \bar{\underline{\underline{\varepsilon}}}^p_{m+1}(\underline{x})\dot{\lambda}_{m+1}(t) = \bar{\underline{\underline{\varepsilon}}}^p(\underline{x})\dot{\lambda}(t) \end{array} \tag{4.113}$$

• **Space problem**:

To calculate the spatial functions of the corrective terms associated to the admissibility problem we must first consider the weak form of the equilibrium equation presented below:

$\forall \underline{v} \in \mathcal{U}^S(\Omega,0) \otimes \mathcal{U}^T(I)$,

$$\int_{\Omega\times I} \rho \underline{\ddot{u}}_{n+1} \cdot \underline{\dot{v}}\, d\Omega dt + \int_{\Omega\times I} \underline{\underline{\sigma}}_{n+1} : \underline{\underline{\varepsilon}}(\underline{\dot{v}})\, d\Omega dt = \int_{\Omega\times I} \rho \underline{f} \cdot \underline{\dot{v}}\, d\Omega dt + \int_{\partial_N\Omega\times I} \underline{f}^N \cdot \underline{\dot{v}}\, dS dt \tag{4.114}$$

By subtracting the equilibrium equations related to the solutions sets $\mathcal{S}_{n+1}$ and $\mathcal{S}_n$ we obtain the equilibrium equation written in function of the corrective quantities as follows:

$\forall \underline{v} \in \mathcal{U}^S(\Omega,0) \otimes \mathcal{U}^T(I)$,

$$\int_{\Omega\times I} \rho \Delta\underline{\ddot{u}}_{n+1} \cdot \underline{\dot{v}}\, d\Omega dt + \int_{\Omega\times I} \Delta\underline{\underline{\sigma}}_{n+1} : \underline{\underline{\varepsilon}}(\underline{\dot{v}})\, d\Omega dt = 0 \tag{4.115}$$

In this case, unlike the isotropic damage case, the kinematic admissibility of the strain correction equation is used, this allows to obtain an equilibrium equation in function of the total corrective strain:

$\forall \underline{\underline{\sigma}}^* \in \mathcal{F}(\Omega,0)\ ,\ \forall \underline{v} \in \mathcal{U}^S(\Omega,0) \otimes \mathcal{U}^T(I)$,

$$\int_{\Omega\times I} \rho \Delta\underline{\dot{u}}_{n+1} \cdot \underline{\ddot{v}}\, d\Omega dt + \int_{\Omega\times I} \Delta\underline{\underline{\dot{\varepsilon}}}_{n+1} : \underline{\underline{\sigma}}^* d\Omega dt = 0 \tag{4.116}$$

By using the strain partition relation $\Delta\underline{\underline{\varepsilon}}_{n+1} = \Delta\underline{\underline{\varepsilon}}^e_{n+1} + \Delta\underline{\underline{\varepsilon}}^p_{n+1}$, with $\Delta\underline{\underline{\varepsilon}}^e_{n+1}$ the elastic deformation term, and introducing it into equation (4.116), we obtain:

$\forall \underline{v} \in \mathcal{U}^S(\Omega,0) \otimes \mathcal{U}^T(I),\ \forall \underline{\underline{\sigma}}^* \in \mathcal{F}(\Omega,0)$,

$$\int_{\Omega\times I} \rho \Delta\underline{\dot{u}}_{n+1} \cdot \underline{\ddot{v}}\, d\Omega dt + \int_{\Omega\times I} \left[\mathbb{K}^{-1} : \Delta\underline{\underline{\dot{\sigma}}}_{n+1} + \Delta\underline{\underline{\dot{\varepsilon}}}^p_{n+1}\right] : \underline{\underline{\sigma}}^* d\Omega dt = 0 \tag{4.117}$$

By replacing $\Delta\underline{\underline{\dot{\varepsilon}}}^p_{n+1}$ by using the descent search direction (4.98) into (4.117) and considering the simplified notation of (4.113) along with $\underline{\underline{\sigma}}^* = \lambda\bar{\underline{\underline{\sigma}}}^*$, we obtain:

$\forall \bar{\underline{v}} \in \mathcal{U}^S(\Omega,0) \otimes \mathcal{U}^T(I),\ \forall \bar{\underline{\underline{\sigma}}}^* \in \mathcal{F}(\Omega,0)$,

$$\int_\Omega \rho < \ddot{\lambda}\dot{\lambda} > \bar{\underline{u}} \cdot \bar{\underline{v}}\, d\Omega + \int_\Omega \underbrace{\left[< \mathbb{H}_\sigma \lambda^2 > : \bar{\underline{\underline{\sigma}}} + < \dot{\lambda}\lambda > \mathbb{K}^{-1} : \bar{\underline{\underline{\sigma}}} - < \underline{\underline{\Delta}}_{n+1}\lambda >\right]}_{\underline{\underline{\varepsilon}}(\bar{\underline{u}})} : \bar{\underline{\underline{\sigma}}}^* d\Omega = 0 \tag{4.118}$$

where we recall $< \cdot >= \int_I (\cdot) dt$.

From the above equation we can define a strain field correction $\underline{\underline{\varepsilon}}(\bar{\underline{u}})$ using an unknown displacement space-field $\bar{\underline{u}}$ defined on Ω and kinematically admissible to zero. Furthermore, from the last equation we can define the following operator:

$$\mathbb{W}^{-1} = \left[< \mathbb{H}_\sigma \lambda^2 > + < \dot{\lambda}\lambda > \mathbb{K}^{-1}\right] \tag{4.119}$$

by using this operator and by denoting:

$$\underline{\underline{\delta}}_{n+1} =< \underline{\underline{\Delta}}_{n+1}\lambda > \tag{4.120}$$

the strain $\underline{\underline{\varepsilon}}(\bar{\underline{u}})$ can be rewritten as:

$$\underline{\underline{\varepsilon}}(\bar{\underline{u}}) = \mathbb{W}^{-1} : \bar{\underline{\underline{\sigma}}} - \underline{\underline{\delta}}_{n+1} \tag{4.121}$$

The above equation allows the determination of the stress space functions as:

$$\bar{\underline{\underline{\sigma}}} = \mathbb{W} : (\bar{\underline{\underline{\varepsilon}}}(\bar{\underline{u}}) + \underline{\underline{\delta}}_{n+1}) \tag{4.122}$$

By using the fact that the space function of the stress tensor $\bar{\underline{\underline{\sigma}}}$ must be admissible to zero as shown in equation (4.115), we can write:

$\forall \bar{\underline{v}} \in \mathcal{U}^S(\Omega, 0)$,

$$\int_\Omega \bar{\underline{\underline{\sigma}}} : \bar{\underline{\underline{\varepsilon}}}(\bar{v}) d\Omega = -\frac{\int_\Omega \rho < \ddot{\lambda}\dot{\lambda} > \bar{\underline{u}} \cdot \bar{\underline{v}}\, d\Omega}{< \lambda\dot{\lambda} >} \tag{4.123}$$

which is finally converted using equation (4.122) into:

$\forall \bar{\underline{v}} \in \mathcal{U}^S(\Omega, 0)$,

$$< \ddot{\lambda}\dot{\lambda} > \int_\Omega \rho \bar{\underline{v}} \cdot \bar{\underline{u}}\, d\Omega + < \lambda\dot{\lambda} > \int_\Omega \bar{\underline{\underline{\varepsilon}}}(\bar{\underline{v}}) : \mathbb{W} : \bar{\underline{\underline{\varepsilon}}}(\bar{\underline{u}}) d\Omega = - < \lambda\dot{\lambda} > \int_\Omega \bar{\underline{\underline{\varepsilon}}}(\bar{\underline{v}}) : \mathbb{W} : \underline{\underline{\delta}}_{n+1} d\Omega \tag{4.124}$$

Equation (4.124) allows the determination of the displacement function $\bar{\underline{u}}$ by using classical FEM. Once the corrective spatial displacement field $\bar{\underline{u}} \in \mathcal{U}^S(\Omega, 0)$ is determined, we compute the spatial corrective strain part $\bar{\underline{\underline{\varepsilon}}}(\bar{\underline{u}})$. The space function of the corrective stress is computed by using the expression (4.122) while the function related to the plastic deformation is computed by using (4.98) as follows:

$$\bar{\underline{\underline{\varepsilon}}}^p(\underline{x}) = \frac{1}{< \lambda\dot{\lambda} >} \left[< \lambda\lambda > \mathbb{H}_\sigma : \bar{\underline{\underline{\sigma}}} - \underline{\underline{\delta}}_{n+1} \right] \tag{4.125}$$

Finally the space function associated to the total deformation is calculated using the strain partition property:

$$\bar{\underline{\underline{\varepsilon}}}(\underline{x}) = \mathbb{K}^{-1} : \bar{\underline{\underline{\sigma}}}(\underline{x}) + \bar{\underline{\underline{\varepsilon}}}^p(\underline{x}) \tag{4.126}$$

Remark: The choice of the operator $\mathbb{H}_\sigma$ used at this step affects the rate of convergence of the method. As exposed at the beginning of this section, the operator used in the descent search direction related to the admissibility problem has the following form:

$$\mathbb{H}_\sigma = c(\mathbb{K})^{-1}$$

with c a constant scalar.

In what follows, it will be shown how the simple choice of the scalar c can improve the convergence of the LATIN-PGD method. In fact, by replacing its expression into the construction of the spatial functions $\bar{\underline{\underline{\sigma}}}_{m+1}(\underline{x})$ and $\bar{\underline{\underline{\varepsilon}}}^p_{m+1}(\underline{x})$ ((4.122) and (4.125) respectively) we have:

$$\bar{\underline{\underline{\sigma}}}_{m+1}(\underline{x}) = \left(< c\lambda^2 > + < \dot{\lambda}\lambda > \right)^{-1} (\mathbb{K}) : (\bar{\underline{\underline{\varepsilon}}}(\bar{\underline{u}}) + \underline{\underline{\delta}}_{n+1}) \tag{4.127}$$

$$\bar{\underline{\underline{\varepsilon}}}^p_{m+1}(\underline{x}) = \frac{1}{< \lambda\dot{\lambda} >} \left[< \lambda\lambda > c\,(\mathbb{K})^{-1} : \bar{\underline{\underline{\sigma}}}_{m+1} - \underline{\underline{\delta}}_{n+1} \right] \tag{4.128}$$

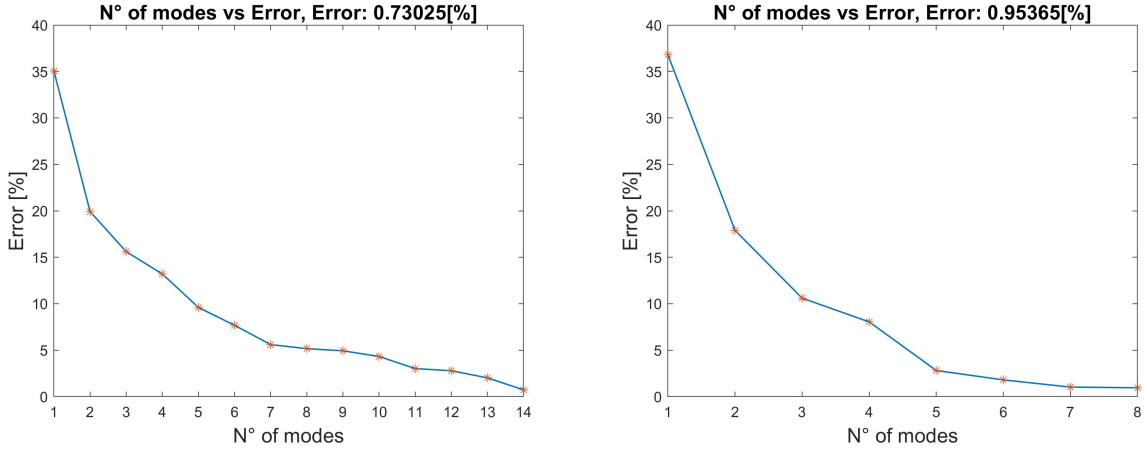
From the above equations it is easy to see that, the smaller the constant c, the more localized is the spatial function associated with the plastic correction, since its construction consists of two contributions, one from the space function $\bar{\underline{\underline{\sigma}}}_{m+1}$ and the other from the function $\underline{\underline{\delta}}_{n+1}$ as seen below:

$$\bar{\underline{\underline{\varepsilon}}}^p_{m+1}(\underline{x}) = \frac{1}{< \lambda\dot{\lambda} >} \left[< \lambda\lambda > c\,(\mathbb{K})^{-1} : \underbrace{\bar{\underline{\underline{\sigma}}}_{m+1}}_{\text{global}} - \underbrace{\underline{\underline{\delta}}_{n+1}}_{\text{local}} \right] \tag{4.129}$$

As can be seen from the above equations, the spatial function related to stress is global since its calculation comes from a FEM resolution, however, the function $\underline{\underline{\delta}}_{n+1}$ is local in space since it comes mainly from the evaluation of the constitutive relation.

The above property is of key importance when dealing with problems with a high degree of localization of the constitutive relation quantities. Indeed, on this situation if we use a classic constant operator as the Hooke's tensor ($c = 1$) we will construct plastic space functions that will have a global behavior rather than a local one. This will produce that the LATIN-PGD solver converges slowly to the desired solution since many iterations will be required just to remove the global plastic deformation that is added in spatial zones where there is actually no plastic deformation on the local solution $\hat{\mathcal{S}}$. Therefore, in order to improve the constant operator $\mathbb{H}_\sigma$ a constant $c = 10^{-4}$ is chosen.

To illustrate the advantages of modifying the constant search direction operator by simply multiplying it by a constant c, a 3D bending beam problem with vertical displacements imposed at both ends (as the numerical tests to be presented in section 4.6.2) is solved. Under these conditions the reference problem is solved by considering a constant $c = 1$ and $c = 10^{-4}$. The plots of LATIN error versus the number of modes is presented in figure 4.10 for both cases.



(a) Error vs N° of modes when $c = 1$.

(b) Error vs N° of modes when $c = 10^{-4}$.

Figure 4.10: LATIN error versus number of modes.

From the previous figure it can be observed that many more modes are needed for the case of $c = 1$ than for the case of $c = 10^{-4}$, where this simple modification allows to obtain better PGD modes. Moreover, a reduction in the computational cost is achieved, the table 4.1 exposes the computational time needed to solve the problem considering both cases.

Operator's factor (c)	Computational time
$c = 1$	5.72 minutes
$c = 10^{-4}$	2.8 minutes

Table 4.1: Computational time with different factor c.

From these results we can observe a $51[\%]$ reduction in the overall computational cost, which desearves to be emphasized considering that only one multiplication factor separates the two operators. Furthermore, the number of LATIN iterations needed to converge are also reduced, where 28 iterations are required with $c = 10^{-4}$, while 60 are needed for $c = 1$, showing that a better operator is built while maintaining its simplicity.

Remark: It should be noted that the same procedure as described previously cannot be applied to the case of isotropic damage. This is due to the fact that the spatial PGD functions used in that case, which correspond to the stress and total deformation, are both global variables (determined directly from a FEM formulation, see (4.52)), making it impossible the application of the above methodology.

• **Temporal problem**:

Once the space functions $\underline{\bar{u}}_{m+1}$, $\underline{\underline{\bar{\sigma}}}_{m+1}$ and $\underline{\underline{\bar{\varepsilon}}}^p_{m+1}$ of the corrective terms (4.110) have been computed, the common temporal function λ_{m+1} is calculated by minimizing the CRE of equation (4.104) as shown below:

$$\{\lambda_{m+1}\} = \underset{\lambda_{m+1}\in\mathcal{U}^T}{\arg\min} \left|\left|\left| \underline{\underline{\bar{\varepsilon}}}^p_{m+1}\dot{\lambda}_{m+1} - \mathbb{H}_\sigma : \underline{\underline{\bar{\sigma}}}_{m+1}\lambda_{m+1} + \underline{\underline{\Delta}}_{n+1} \right|\right|\right|^2_{\mathbb{H}_\sigma^{-1}} \tag{4.130}$$

In what follows we drop out the mode index to simplify the notations following the expressions in (4.113) and we minimize equation (4.130) with respect to λ, giving:

$\forall \delta\lambda \in \mathcal{U}^T$,

$$\int_{\Omega\times I} (\underline{\underline{\bar{\varepsilon}}}^p \delta\dot{\lambda} - \mathbb{H}_\sigma : \underline{\underline{\bar{\sigma}}}\delta\lambda) : \mathbb{H}_\sigma^{-1} : (\underline{\underline{\bar{\varepsilon}}}^p \dot{\lambda} - \mathbb{H}_\sigma : \underline{\underline{\bar{\sigma}}}\lambda + \underline{\underline{\Delta}}_{n+1})\, d\Omega dt = 0 \tag{4.131}$$

By expanding this last equation, we obtain:

$\forall \delta\lambda \in \mathcal{U}^T$,

$$\begin{aligned}
&\int_I \delta\dot{\lambda}\dot{\lambda} dt \int_\Omega (\underline{\underline{\bar{\varepsilon}}}^p : \mathbb{H}_\sigma^{-1} : \underline{\underline{\bar{\varepsilon}}}^p)\, d\Omega - \int_I \delta\dot{\lambda}\lambda \int_\Omega \underline{\underline{\bar{\varepsilon}}}^p : \underline{\underline{\bar{\sigma}}}\, d\Omega dt + \int_I \delta\dot{\lambda} \int_\Omega (\underline{\underline{\bar{\varepsilon}}}^p : \mathbb{H}_\sigma^{-1} : \underline{\underline{\Delta}}_{n+1})\, d\Omega dt \\
&- \int_I \delta\lambda\dot{\lambda} \int_\Omega \underline{\underline{\bar{\sigma}}} : \underline{\underline{\bar{\varepsilon}}}^p\, d\Omega dt + \int_I \delta\lambda\lambda \int_\Omega (\mathbb{H}_\sigma : \underline{\underline{\bar{\sigma}}}) : \underline{\underline{\bar{\sigma}}}\, d\Omega dt \\
&- \int_I \delta\lambda \int_\Omega \underline{\underline{\bar{\sigma}}} : \underline{\underline{\Delta}}_{n+1}\, d\Omega dt = 0
\end{aligned} \tag{4.132}$$

By denoting the constants A^{11}, A^{10}, A^{01}, A^{00} and the temporal functions $D^1(t)$ and $D^0(t)$ as follows:

$$\begin{aligned}
A^{11} &= \int_\Omega \underline{\underline{\bar{\varepsilon}}}^p : \mathbb{H}_\sigma^{-1} : \underline{\underline{\bar{\varepsilon}}}^p\, d\Omega \\
A^{10} &= -\int_\Omega \underline{\underline{\bar{\varepsilon}}}^p : \underline{\underline{\bar{\sigma}}}\, d\Omega \qquad\qquad D^1(t) = -\int_\Omega (\underline{\underline{\bar{\varepsilon}}}^p : \mathbb{H}_\sigma^{-1} : \underline{\underline{\Delta}}_{n+1})\, d\Omega \\
A^{01} &= -\int_\Omega \underline{\underline{\bar{\sigma}}} : \underline{\underline{\bar{\varepsilon}}}^p\, d\Omega \qquad\qquad D^0(t) = \int_\Omega \underline{\underline{\bar{\sigma}}} : \underline{\underline{\Delta}}_{n+1}\, d\Omega \\
A^{00} &= \int_\Omega (\mathbb{H}_\sigma : \underline{\underline{\bar{\sigma}}}) : \underline{\underline{\bar{\sigma}}}\, d\Omega
\end{aligned} \quad , \tag{4.133}$$

the expression (4.132) can be simplified to:

$\forall \delta\lambda \in \mathcal{U}^T$,

$$\int_I \delta\dot{\lambda}\dot{\lambda}A^{11} + \delta\dot{\lambda}\lambda A^{10} + \delta\lambda\dot{\lambda}A^{01} + \delta\lambda\lambda A^{00} = \int_I \delta\dot{\lambda}D^1(t) + \delta\lambda D^0(t) \tag{4.134}$$

By applying the TDGM in the same way as presented for the isotropic damage case, the discretization of expression (4.134) applied on the time element "k" gives:

$$\left(\underline{\underline{\boldsymbol{Q}}}^{[k]} + \underline{\underline{\boldsymbol{\mathcal{L}}}}^{[k]}\right) \underline{\boldsymbol{\lambda}}^{[k]} = \underline{\underline{\boldsymbol{\mathcal{R}}}}^{[k]}\, \underline{\boldsymbol{\lambda}}^{[k-1]} + \underline{\boldsymbol{f}}^{[k]} \tag{4.135}$$

where $\underline{\boldsymbol{\lambda}}^{[k]}$ corresponds to the nodal values of the time function at time step k, and where the elementary discretized matrix is given by:

$$\underline{\underline{\boldsymbol{Q}}}^{[k]} = \int_{\check{I}_k} \left[\underline{\dot{\psi}}^{[k]}(t) \otimes \underline{\dot{\psi}}^{[k]}(t) A^{11} + \underline{\dot{\psi}}^{[k]}(t) \otimes \underline{\psi}^{[k]}(t) A^{10} + \underline{\psi}^{[k]}(t) \otimes \underline{\dot{\psi}}^{[k]}(t) A^{01} + \underline{\psi}^{[k]}(t) \otimes \underline{\psi}^{[k]}(t) A^{00} \right] dt \tag{4.136}$$

and the discretized vector by:

$$\underline{\boldsymbol{f}}^{[k]} = \int_{\check{I}_k} \left[\underline{\dot{\psi}}^{[k]}(t) D^1(t) + \underline{\psi}^{[k]}(t) D^0(t) \right] dt \tag{4.137}$$

In addition, the matrices that allows to weakly impose the continuity between the temporal intervals are given by:

$$\begin{aligned} \underline{\underline{\boldsymbol{\mathcal{L}}}}^{[k]} &= 1.1 \max(\underline{\underline{\boldsymbol{Q}}}^{[k]}) \underline{\psi}^{[k]}(t_{k-1}) \otimes \underline{\psi}^{[k]}(t_{k-1}^{+}) \\ \underline{\underline{\boldsymbol{\mathcal{R}}}}^{[k]} &= 1.1 \max(\underline{\underline{\boldsymbol{Q}}}^{[k]}) \underline{\psi}^{[k]}(t_{k-1}) \otimes \underline{\psi}^{[k-1]}(t_{k-1}^{-}) \end{aligned} \tag{4.138}$$

4.4.2.1.2 Space and time resolution associated to the internal variables

The enrichment step associated to the internal variables follows the same rules as those outlined above for the admissibility problem, i.e., the alternate resolution of a space and time problem by a fixed point strategy. In this case it is also necessary to control the number of iterations of the fixed-point algorithm by defining a stagnation quantity, which is determined for each internal variable. This implies once again the re-scaling of the spatial and temporal functions, which is done by defining the norms associated to the space PGD function of kinematic and isotropic hardening as:

$$\begin{aligned} c_\alpha &= \left\| \underline{\bar{\underline{\alpha}}}_{m+1} \right\|_\Omega \\ c_{\bar{r}} &= \| \bar{\bar{r}}_{m+1} \|_\Omega \end{aligned} \tag{4.139}$$

From the two defined constants of equation (4.139), the PGD functions related to kinematic hardening are re-scaled as:

$$\underline{\bar{\underline{\alpha}}}_{m+1} \leftarrow \underline{\bar{\underline{\alpha}}}_{m+1} / c_\alpha \qquad , \qquad \begin{aligned} \lambda^{\alpha}_{m+1} &\leftarrow \lambda^{\alpha}_{m+1}\, c_\alpha \\ \dot{\lambda}^{\alpha}_{m+1} &\leftarrow \dot{\lambda}^{\alpha}_{m+1}\, c_\alpha \end{aligned}$$

while the ones related to isotropic hardening by:

$$\bar{\bar{r}}_{m+1} \leftarrow \bar{\bar{r}}_{m+1} / c_{\bar{r}} \qquad , \qquad \begin{aligned} \lambda^{\bar{r}}_{m+1} &\leftarrow \lambda^{\bar{r}}_{m+1}\, c_{\bar{r}} \\ \dot{\lambda}^{\bar{r}}_{m+1} &\leftarrow \dot{\lambda}^{\bar{r}}_{m+1}\, c_{\bar{r}} \end{aligned}$$

Once the PGD functions re-scaled, the definition of the stagnation indicators for each internal variable are given below:

$$\zeta^{\alpha}_{PGD} = \frac{\left\| |(\lambda^{\alpha}_{m+1})^{(i)}| - |(\lambda^{\alpha}_{m+1})^{(i-1)}| \right\|_I}{\left\| |(\lambda^{\alpha}_{m+1})^{(i)}| + |(\lambda^{\alpha}_{m+1})^{(i-1)}| \right\|_I} \tag{4.140}$$

$$\zeta^{\bar{r}}_{PGD} = \frac{\left\| |(\lambda^{\bar{r}}_{m+1})^{(i)}| - |(\lambda^{\bar{r}}_{m+1})^{(i-1)}| \right\|_I}{\left\| |(\lambda^{\bar{r}}_{m+1})^{(i)}| + |(\lambda^{\bar{r}}_{m+1})^{(i-1)}| \right\|_I} \tag{4.141}$$

The enrichment step (fixed point iterations) is applied separately for the kinematic and isotropic hardening quantities and is stopped when the stagnation quantity is less than a given threshold or a maximum number of iterations is reached in the same way as for the admissibility quantities.

Since the internal variables does not require to verify external boundary conditions, its space and temporal functions are simply determined by minimizing the CRE presented in equations (4.105) and (4.106) for the case of kinematic and isotropic hardening respectively.

In the following lines the spatial and temporal problem associated with the enrichment stage is detailed, where the index related to the PGD mode is left out, in order to alleviate the notation, writing:

$$\begin{aligned}\Delta\dot{\underline{\underline{\alpha}}}_{n+1} &= \underline{\underline{\bar{\alpha}}}_{m+1}(\underline{x})\dot{\lambda}^{\alpha}_{m+1}(t) = \underline{\underline{\bar{\alpha}}}(\underline{x})\dot{\lambda}^{\alpha}(t)\\ \Delta\dot{\bar{\bar{r}}}_{n+1} &= \bar{\bar{r}}_{m+1}(\underline{x})\dot{\lambda}^{\bar{r}}_{m+1}(t) = \bar{\bar{r}}(\underline{x})\dot{\lambda}^{\bar{r}}(t)\end{aligned} \tag{4.142}$$

• **Space problem**:

The calculation of the spatial functions related to the PGD approximation of the internal variables are simply determined by solving their descent search direction, obtained in section 4.4.2, which are recalled below for convenience:

$$\begin{aligned}\Delta\dot{\underline{\underline{\alpha}}}_{n+1} + \mathbb{H}_{\beta} : (C\Delta\underline{\underline{\alpha}}_{n+1}) + \underline{\underline{\Delta}}^{\alpha}_{n+1} = 0\\ \Delta\dot{\bar{\bar{r}}}_{n+1} + H_{\bar{R}}R_{\infty}\Delta\bar{r}_{n+1} + \Delta^{\bar{r}}_{n+1} = 0\end{aligned}$$

with the residual functions given by:

$$\underline{\underline{\Delta}}^{\alpha}_{n+1} = \dot{\underline{\underline{\alpha}}}_{n} - \hat{\dot{\underline{\underline{\alpha}}}}_{n+1/2} \tag{4.143}$$

$$\Delta^{\bar{r}}_{n+1} = \dot{\bar{\bar{r}}}_{n} - \hat{\dot{\bar{\bar{r}}}}_{n+1/2} \tag{4.144}$$

By using the PGD approximations and simplifications of equation (4.142) we have:

$$\underline{\underline{\bar{\alpha}}}\dot{\lambda}^{\alpha} + \mathbb{H}_{\beta} : C\underline{\underline{\bar{\alpha}}}\lambda^{\alpha} + \underline{\underline{\Delta}}^{\alpha}_{n+1} = 0 \tag{4.145}$$

$$\bar{\bar{r}}\dot{\lambda}^{\bar{r}} + H_{\bar{R}}R_{\infty}\bar{\bar{r}}\lambda^{\bar{r}} + \Delta^{\bar{r}}_{n+1} = 0 \tag{4.146}$$

As mentioned previously, no admissibility conditions are imposed to the internal variables, therefore the space functions can be simply calculated after integrating the above expressions in time as follows:

$$\underline{\underline{\bar{\alpha}}} = \left[(\int_I \dot{\lambda}^{\alpha}dt) + \mathbb{H}_{\beta}C(\int_I \lambda^{\alpha}dt)\right]^{-1} : \left(-\int_I \underline{\underline{\Delta}}^{\alpha}_{n+1}dt\right) \tag{4.147}$$

$$\bar{\bar{r}} = \left[(\int_I \dot{\lambda}^{\bar{r}}dt) + H_{\bar{R}}R_{\infty}(\int_I \lambda^{\bar{r}}dt)\right]^{-1} \left(-\int_I \Delta^{\bar{r}}_{n+1}dt\right) \tag{4.148}$$

• **Time problem**:

The temporal functions of the PGD decomposition on the other hand are calculated by minimizing the constitutive relation errors related to kinematic and isotropic hardening((4.105) and (4.106) respectively) writing:

$$\{\lambda^{\alpha}\} = \underset{\lambda^{\alpha}\in\mathcal{U}^T}{\arg\min} \left|\left|\left|\underline{\underline{\bar{\alpha}}}\dot{\lambda}^{\alpha} + \mathbb{H}_{\beta} : C\underline{\underline{\bar{\alpha}}}\lambda^{\alpha} + \underline{\underline{\Delta}}^{\alpha}_{n+1}\right|\right|\right|^2_{\mathbb{H}^{-1}_{\beta}} \tag{4.149}$$

$$\{\lambda^{\bar{r}}\} = \underset{\lambda^{\bar{r}}\in\mathcal{U}^T}{\arg\min} \left|\left|\left|\bar{\bar{r}}\dot{\lambda}^{\bar{r}} + H_{\bar{R}}R_{\infty}\bar{\bar{r}}\lambda^{\bar{r}} + \Delta^{\bar{r}}_{n+1}\right|\right|\right|^2_{H^{-1}_{\bar{R}}} \tag{4.150}$$

As can be seen, the determination of the time functions follows the same steps as those associated with the admissibility problem, therefore, the application of the TDGM for this case is omitted herein but its presentation is detailed in appendix C.

4.4.2.2 Preliminary step

As for the case of isotropic damage, once a new mode is determined at the enrichment step, the PGD decomposition is improved by actualizing the time PGD functions at the preliminary step. Since in elasto-visco-plasticity we have the low-rank approximation of the admissibility terms and the ones related to the internal variables, the preliminary step is applied to both of them. But this imposes a limitation, indeed the preliminary step is an iterative process that must be stopped when reaching a given stagnation limit; this stagnation can't be defined over the admissibility and the internal variables terms at the same time, and only one of them must be chosen for the definition of the stagnation indicator. The one selected herein corresponds to the admissibility terms, writing:

$$\zeta_{up} = \max \left\| 0.5 \left(\frac{|\Delta \underline{\boldsymbol{\Lambda}}|}{|\underline{\boldsymbol{\Lambda}}| + |\Delta \underline{\boldsymbol{\Lambda}}|} \right) \right\|_2 \tag{4.151}$$

where we recall the terms $\underline{\boldsymbol{\Lambda}}$ and $\Delta\underline{\boldsymbol{\Lambda}}$ which correspond to the matrices that contains all the discretized time PGD functions and their respective correction for the admissibility quantities computed at this actualization step:

$$\underline{\boldsymbol{\Lambda}} = \begin{bmatrix} \underline{\boldsymbol{\lambda}}_1 \\ \underline{\boldsymbol{\lambda}}_2 \\ \vdots \\ \underline{\boldsymbol{\lambda}}_{m+1} \end{bmatrix} \quad , \quad \Delta\underline{\boldsymbol{\Lambda}} = \begin{bmatrix} \Delta\underline{\boldsymbol{\lambda}}_1 \\ \Delta\underline{\boldsymbol{\lambda}}_2 \\ \vdots \\ \Delta\underline{\boldsymbol{\lambda}}_{m+1} \end{bmatrix}$$

The stagnation of equation (4.151) is computed each time the actualization of the temporal functions is done. Once the preliminary step is completed, the final approximation of the PGD decomposition related to the admissibility quantities can be rewritten as:

$$\begin{aligned}
\underline{u}_{n+1}(\underline{x},t) &= \sum_{i=1}^{m+1} \bar{\underline{u}}_i(\underline{x}) \left[\lambda_i(t) + \Delta\lambda_i(t)\right] + \underline{u}_0(\underline{x},t) \\
\underline{\underline{\varepsilon}}_{n+1}(\underline{x},t) &= \sum_{i=1}^{m+1} \bar{\underline{\underline{\varepsilon}}}_i(\underline{x}) \left[\lambda_i(t) + \Delta\lambda_i(t)\right] + \underline{\underline{\varepsilon}}_0(\underline{x},t) \\
\underline{\underline{\sigma}}_{n+1}(\underline{x},t) &= \sum_{i=1}^{m+1} \bar{\underline{\underline{\sigma}}}_i(\underline{x}) \left[\lambda_i(t) + \Delta\lambda_i(t)\right] + \underline{\underline{\sigma}}_0(\underline{x},t) \\
\dot{\underline{\underline{\varepsilon}}}^p_{n+1}(\underline{x},t) &= \sum_{i=1}^{m+1} \bar{\underline{\underline{\varepsilon}}}^p_i(\underline{x}) \left[\dot{\lambda}_i(t) + \Delta\dot{\lambda}_i(t)\right]
\end{aligned} \tag{4.152}$$

The update of the time functions associated with the internal variables, unlike the case of the admissibility quantities, imposes a major limitation, which lies in the poor conditioning found in numerical tests. To solve this ill-conditioning and at the same time improve the PGD temporal functions, only the last mode is updated, i.e:

$$\begin{aligned}
\underline{\underline{\alpha}}_{n+1}(\underline{x},t) &= \sum_{i=1}^{m} \bar{\underline{\underline{\alpha}}}_i(\underline{x}) \lambda_i^{\alpha}(t) + \bar{\underline{\underline{\alpha}}}_{m+1}(\underline{x}) \left[\lambda_{m+1}^{\alpha}(t) + \Delta\lambda_{m+1}^{\alpha}(t)\right] \\
\bar{r}_{n+1}(\underline{x},t) &= \sum_{i=1}^{m} \bar{\bar{r}}_i(\underline{x}) \lambda_i^{\bar{r}}(t) + \bar{\bar{r}}_{m+1}(\underline{x}) \left[\lambda_{m+1}^{\bar{r}}(t) + \Delta\lambda_{m+1}^{\bar{r}}(t)\right]
\end{aligned} \tag{4.153}$$

4.4.2.2.1 Time functions associated to the admissibility quantities

The corrective temporal functions are determined such that the constitutive relation error is minimized, in this sense, by injecting the PGD approximations of (4.152) and introducing it into the CRE (4.104) we

obtain:

$\forall i \in [1, ..., m+1]$,

$$\{\Delta\lambda_i\} = \underset{\{\Delta\lambda_i\}_{i=1}^{m+1} \in \mathcal{U}^T}{\arg\min} \left| \left| \left| \sum_{i=1}^{m+1} \underline{\underline{\bar{\varepsilon}}}_i^p \Delta\dot{\lambda}_i - \mathbb{H}_\sigma : \sum_{i=1}^{m+1} \underline{\underline{\bar{\sigma}}}_i \Delta\lambda_i + \underline{\underline{\Delta}}_{n+1} \right| \right| \right|_{\mathbb{H}_\sigma^{-1}}^2 \tag{4.154}$$

The minimization problem of expression (4.154) gives:

$\forall \delta\Delta\lambda_i \in \mathcal{U}^T,\ \forall i \in [1, ..., m+1]\,,$

$$\int_{\Omega \times I} \left(\sum_{i=1}^{m+1} \underline{\underline{\bar{\varepsilon}}}_i^p \delta\Delta\dot{\lambda}_i - \mathbb{H}_\sigma : \sum_{i=1}^{m+1} \underline{\underline{\bar{\sigma}}}_i \delta\Delta\lambda_i \right) : \mathbb{H}_\sigma^{-1} : \left(\sum_{j=1}^{m+1} \underline{\underline{\bar{\varepsilon}}}_j^p \Delta\dot{\lambda}_j - \mathbb{H}_\sigma : \sum_{j=1}^{m+1} \underline{\underline{\bar{\sigma}}}_j \Delta\lambda_j + \underline{\underline{\Delta}}_{n+1} \right) = 0 \tag{4.155}$$

which can be simplified in:

$\forall \delta\Delta\lambda_i \in \mathcal{U}^T,\ \forall i \in [1, ..., m+1]\,,$

$$\int_I \sum_{i=1}^{m+1} \sum_{j=1}^{m+1} \delta\Delta\dot{\lambda}_i \left(\Delta\dot{\lambda}_j A_{ij}^{11} + \Delta\lambda_j A_{ij}^{10} \right) + \delta\Delta\lambda_i \left(\Delta\dot{\lambda}_j A_{ij}^{01} + \Delta\lambda_j A_{ij}^{00} \right) dt = \int_I \sum_{i=1}^{m+1} \delta\Delta\dot{\lambda}_i D_i^1(t) + \delta\Delta\lambda_i D_i^0(t)\ dt \tag{4.156}$$

with the different constants and temporal functions terms given by:

$\forall (i,j) \in [1, ..., m+1]\,,$

$$\begin{aligned} A_{ij}^{11} &= \int_\Omega \underline{\underline{\bar{\varepsilon}}}_i^p : \mathbb{H}_\sigma^{-1} : \underline{\underline{\bar{\varepsilon}}}_j^p \ d\Omega \\ A_{ij}^{10} &= -\int_\Omega \underline{\underline{\bar{\varepsilon}}}_i^p : \underline{\underline{\bar{\sigma}}}_j \ d\Omega \\ A_{ij}^{01} &= -\int_\Omega \underline{\underline{\bar{\sigma}}}_i : \underline{\underline{\bar{\varepsilon}}}_j^p \ d\Omega \\ A_{ij}^{00} &= \int_\Omega (\mathbb{H}_\sigma : \underline{\underline{\bar{\sigma}}}_i) : \underline{\underline{\bar{\sigma}}}_j \ d\Omega \end{aligned} \quad , \quad \begin{aligned} D_i^1(t) &= -\int_\Omega (\underline{\underline{\bar{\varepsilon}}}_i^p : \mathbb{H}_\sigma^{-1} : \underline{\underline{\Delta}})\ d\Omega \\ D_i^0(t) &= \int_\Omega \underline{\underline{\bar{\sigma}}}_i : \underline{\underline{\Delta}}\ d\Omega \end{aligned} \tag{4.157}$$

Discretizing once again the above system of equations using the TDGM, the discrete form of equation (4.156) on a time element "k" is given by:

$$\left(\underline{\underline{\boldsymbol{Q}}}_{up}^{[k]} + \underline{\underline{\boldsymbol{\mathcal{L}}}}_{up}^{[k]} \right) \Delta\underline{\boldsymbol{\Lambda}}^{[k]} = \underline{\underline{\boldsymbol{\mathcal{R}}}}_{up}^{[k]} \Delta\underline{\boldsymbol{\Lambda}}^{[k-1]} + \underline{\boldsymbol{f}}_{up}^{[k]} \tag{4.158}$$

where the full expression of the matrices are given by:

$$\underline{\underline{\boldsymbol{Q}}}_{up}^{[k]} = \begin{bmatrix} \underline{\underline{\boldsymbol{Q}}}_{11}^{[k]} & \underline{\underline{\boldsymbol{Q}}}_{12}^{[k]} & \cdots & \underline{\underline{\boldsymbol{Q}}}_{1\ m+1}^{[k]} \\ \underline{\underline{\boldsymbol{Q}}}_{21}^{[k]} & \ddots & & \\ \vdots & & & \\ \underline{\underline{\boldsymbol{Q}}}_{m+1\ 1}^{[k]} & & & \underline{\underline{\boldsymbol{Q}}}_{m+1\ m+1}^{[k]} \end{bmatrix} \quad , \quad \underline{\boldsymbol{f}}_{up}^{[k]} = \begin{bmatrix} \underline{\boldsymbol{f}}_1^{[k]} \\ \underline{\boldsymbol{f}}_2^{[k]} \\ \vdots \\ \\ \underline{\boldsymbol{f}}_{m+1}^{[k]} \end{bmatrix}$$

with:

$\forall (i,j) \in [1, ..., m+1]\,,$

$$\underline{\underline{\boldsymbol{Q}}}_{ij}^{[k]} = \int_{\check{I}_k} \left(\underline{\dot{\psi}}^{[k]}(t) \otimes \underline{\dot{\psi}}^{[k]}(t) A_{ij}^{11} + \underline{\dot{\psi}}^{[k]}(t) \otimes \underline{\psi}^{[k]}(t) A_{ij}^{10} + \underline{\psi}^{[k]}(t) \otimes \underline{\dot{\psi}}^{[k]}(t) A_{ij}^{01} + \underline{\psi}^{[k]}(t) \otimes \underline{\psi}^{[k]}(t) A_{ij}^{00} \right) dt \tag{4.159}$$

$$\underline{\boldsymbol{f}}_i^{[k]} = \int_{\breve{I}_k} \left(\underline{\dot{\psi}}^{[k]}(t) D_i^1(t) + \underline{\psi}^{[k]}(t) D_i^0(t) \right) dt \tag{4.160}$$

and the matrices that allows the weak imposition of the initial conditions over all the temporal functions are given by:

$$\underline{\underline{\boldsymbol{\mathcal{L}}}}_{up}^{[k]} = \begin{bmatrix} \underline{\underline{\boldsymbol{\mathcal{L}}}}_{11}^{[k]} & \underline{\underline{\boldsymbol{\mathcal{L}}}}_{12}^{[k]} & \cdots & \underline{\underline{\boldsymbol{\mathcal{L}}}}_{1\ m+1}^{[k]} \\ \underline{\underline{\boldsymbol{\mathcal{L}}}}_{21}^{[k]} & \ddots & & \\ \vdots & & & \\ \underline{\underline{\boldsymbol{\mathcal{L}}}}_{m+1\ 1}^{[k]} & & & \underline{\underline{\boldsymbol{\mathcal{L}}}}_{m+1\ m+1}^{[k]} \end{bmatrix} \quad , \quad \underline{\underline{\boldsymbol{\mathcal{R}}}}_{up}^{[k]} = \begin{bmatrix} \underline{\underline{\boldsymbol{\mathcal{R}}}}_{11}^{[k]} & \underline{\underline{\boldsymbol{\mathcal{R}}}}_{12}^{[k]} & \cdots & \underline{\underline{\boldsymbol{\mathcal{R}}}}_{1\ m+1}^{[k]} \\ \underline{\underline{\boldsymbol{\mathcal{R}}}}_{21}^{[k]} & \ddots & & \\ \vdots & & & \\ \underline{\underline{\boldsymbol{\mathcal{R}}}}_{m+1\ 1}^{[k]} & & & \underline{\underline{\boldsymbol{\mathcal{R}}}}_{m+1\ m+1}^{[k]} \end{bmatrix}$$

with:

$\forall (i,j) \in [1, ..., m+1]\,,$

$$\underline{\underline{\boldsymbol{\mathcal{L}}}}_{ij}^{[k]} = 1.1 \max(\underline{\underline{\boldsymbol{Q}}}_{ij}^{[k]}) \underline{\psi}^{[k]}(t_{k-1}) \otimes \underline{\psi}^{[k]}(t_{k-1}^+) \tag{4.161}$$

$$\underline{\underline{\boldsymbol{\mathcal{R}}}}_{ij}^{[k]} = 1.1 \max(\underline{\underline{\boldsymbol{Q}}}_{ij}^{[k]}) \underline{\psi}^{[k]}(t_{k-1}) \otimes \underline{\psi}^{[k-1]}(t_{k-1}^-) \tag{4.162}$$

Remark: The resolution technique described above allows to obtain the time functions element by element of the time FEM discretization in an incremental way. This fact is of great importance since it overcomes a major limitation when solving the temporal functions by minimizing the constitutive relation error when considering an elasto-visco-plastic behavior. This problem, which was first exposed in Jean-Charles Passieux's thesis [Passieux, 2008], shows that the minimization process of equation (4.130) generates a system of differential equations, which requires as a condition for its resolution the knowledge of the time function at the end of the interval, making its resolution non-trivial. This limitation is avoided using TDGM, where the time functions are obtained element by element due to the minimization of the functional in discontinuous time intervals, where the continuity of the function is weakly imposed from one time interval to the other. This incremental resolution strategy allows to decrease the computational cost of the preliminary step when long duration excitations are considered.

4.4.2.2.2 Time functions associated to internal variables

As exposed at the beginning of the section, due to bad numerical conditioning when updating a pack of temporal PGD modes associated to the internal variables, only the last PGD mode is chosen to be actualized, this is:

$$\lambda_{m+1}^{\alpha} \leftarrow \lambda_{m+1}^{\alpha} + \Delta\lambda_{m+1}^{\alpha} \tag{4.163}$$

$$\lambda_{m+1}^{\bar{r}} \leftarrow \lambda_{m+1}^{\bar{r}} + \Delta\lambda_{m+1}^{\bar{r}} \tag{4.164}$$

with the first m corrective functions set equals to 0:

$\forall i \in [1, ..., m],$

$$\begin{aligned} \Delta\lambda_i^{\alpha} &= 0 \\ \Delta\lambda_i^{\bar{r}} &= 0 \end{aligned} \tag{4.165}$$

By considering the constitutive relation error on the internal variables equations (4.105) and (4.106), and

introducing the above considerations we obtain:

$$\{\Delta\lambda^{\alpha}_{m+1}\} = \underset{\Delta\lambda^{\alpha}_{m+1}\in\mathcal{U}^T}{\arg\min} \left|\left|\left|\underline{\underline{\bar{\alpha}}}_{m+1}\Delta\dot{\lambda}^{\alpha}_{m+1} + \mathbb{H}_{\beta} : C\underline{\underline{\bar{\alpha}}}_{m+1}\Delta\lambda^{\alpha}_{m+1} + \underline{\underline{\Delta}}^{\alpha}_{n+1}\right|\right|\right|^2_{\mathbb{H}^{-1}_{\beta}} \tag{4.166}$$

$$\{\Delta\lambda^{\bar{r}}_{m+1}\} = \underset{\Delta\lambda^{\bar{r}}_{m+1}\in\mathcal{U}^T}{\arg\min} \left|\left|\left|\bar{\bar{r}}_{m+1}\Delta\dot{\lambda}^{\bar{r}}_{m+1} + H_{\bar{R}}R_{\infty}\bar{\bar{r}}_{m+1}\Delta\lambda^{\bar{r}}_{m+1} + \Delta^{\bar{r}}_{n+1}\right|\right|\right|^2_{H^{-1}_{\bar{R}}} \tag{4.167}$$

From the above equations, it can be noted that the determination of the corrective time functions for the case of isotropic and kinematic hardening verifies equations similar to those presented in the enrichment step (see section 4.4.2.1.2). From the above it can be concluded that their determination follows the same steps explained in the appendix C, and consequently the detailed presentation for obtaining the corrective time functions is omitted in this section.

4.4.3 LATIN error indicator

Finally, the LATIN error defined for the elasto-visco-plasticity case is simply considered as follows:

$$\xi_{n+1} = 100\sqrt{\left(\frac{\left|\left|\left|\underline{\underline{\dot{\varepsilon}}}^{p}_{n+1} - \underline{\underline{\hat{\dot{\varepsilon}}}}^{p}_{n+1/2}\right|\right|\right|^2}{\left|\left|\left|\underline{\underline{\dot{\varepsilon}}}^{p}_{n+1}\right|\right|\right|^2} + \frac{\left|\left|\left|\underline{\underline{\sigma}}_{n+1} - \underline{\underline{\hat{\sigma}}}_{n+1/2}\right|\right|\right|^2}{\left|\left|\left|\underline{\underline{\sigma}}_{n+1}\right|\right|\right|^2}\right)[\%]} \tag{4.168}$$

where we recall the norm $\left|\left|\left|\underline{\underline{\bullet}}\right|\right|\right|^2 = \int_{\Omega\times I}(\underline{\underline{\bullet}}) : (\underline{\underline{\bullet}})\, d\Omega dt$.

On the above definition the main variables of interest considered for the definition of the error are the stress $\underline{\underline{\sigma}}$ and the plastic deformation $\underline{\underline{\dot{\varepsilon}}}^{p}$. The internal variables are not considered since its effect is indirectly included in the plastic deformation.

4.5 Synthesis on the LATIN-PGD solver

The LATIN method along with the model reduction technique PGD was presented for solving quasi-fragile isotropic damage concrete and elasto-visco-plasticity problems. The methodology is very similar on both cases, nevertheless, technical implementation details differ.

The key idea of the LATIN-PGD methodology consists in the iterative enrichment of an initial elastic solution (which is calculated in dynamics in this thesis) by an alternating process that consists in two stages, the local and the global ones. On the local step the nonlinear constitutive relation is evaluated, where the calculations are done over the whole space-time integration points. On the global step a linear problem is solved by a low-rank approximation, whose main objective is to compute a corrective term in order to enrich the elastic solution with the objective of minimizing the distance between the local and global space solutions ($\hat{\mathcal{S}}$ and $\mathcal{S}$ respectively). The solver converges when the distance between these two spaces are less than a given threshold [Ladevèze, 1999].

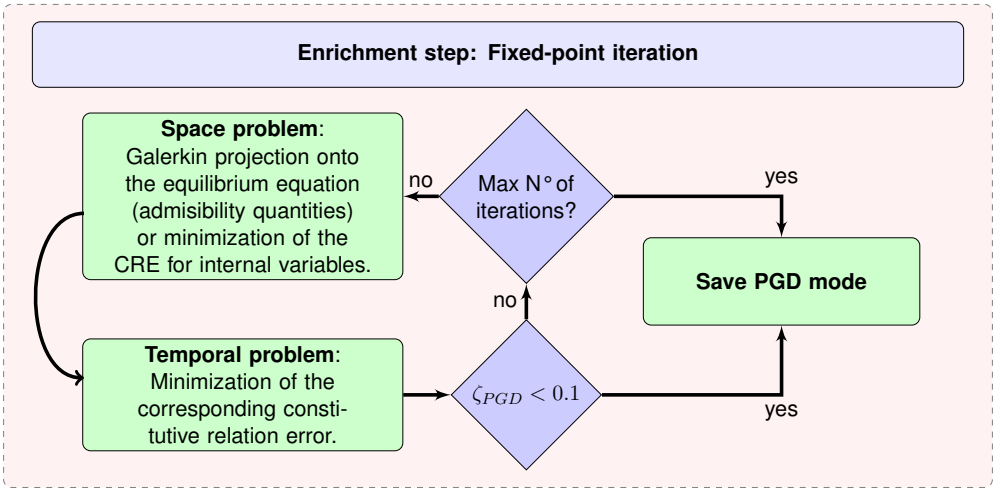


Figure 4.11: Enrichment step, iteratively resolution strategy.

The resolution of the local stage is straightforward, however the global stage is solved in two steps, the enrichment and the preliminary steps. On the enrichment step, a rank one PGD decomposition is added to the whole approximation, where the space functions related to the admissibility problem (displacement, stress, strain, etc) are determined by a Galerkin projection into the equilibrium equation and the temporal functions by minimizing a constitutive relation error (CRE) defined on the search directions and differently for each material considered [Passieux, 2008]. This last is modified for the internal variables of elato-visco-plasticity where no boundary conditions must be verified and therefore the space-time functions are obtained by minimizing their CRE only. The space and time problems are solved iteratively using a fixed-point strategy and the process stops if a certain stagnation threshold ζ_{PGD} is reached or when a maximum number of iterations is surpassed. The diagram in figure 4.11 summarizes this process.

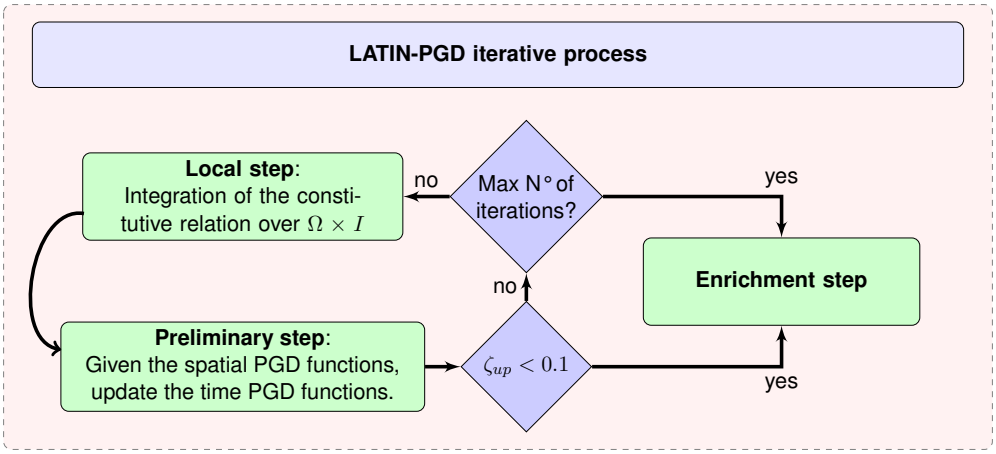


Figure 4.12: LATIN-PGD iterative process.

On the other hand, the preliminary step consists in updating the temporal functions of the already computed PGD approximation. This process is applied after each enrichment step and allows to increase the rate of convergence of the solver while maintaining a low number of PGD modes. The diagram of figure 4.12 summarizes this procedure.

The temporal functions are updated until a given stagnation threshold is reached. Once the preliminary step is done, a new PGD mode is determined on the enrichment step, continuing this process until a

global convergence criterion on the LATIN error $\xi_{n+1} < \xi_{\text{lim}}$ is met. In practice, the error threshold ξ_{lim} is chosen between 0.1 and 2 $[\%]$ to ensure good results.

Algorithm 1 presents a global synthesis of the LATIN-PGD methodology.

Algorithm 1: LATIN-PGD algorithm in low-frequency dynamics.

Input : $\underline{\boldsymbol{u}}_0,\ \underline{\dot{\boldsymbol{u}}}_0, \underline{\underline{\boldsymbol{M}}}\ ,\ \underline{\underline{\boldsymbol{D}}}\ ,\ \underline{\underline{\boldsymbol{K}}},\ (\underline{\boldsymbol{f}}_k)_{k=1}^{N_T},\ (\underline{\boldsymbol{u}}_k^D)_{k=1}^{N_T},\ N_T$ (Discretized elements in time)

Output : $\bar{\mathcal{S}}$: Solution set containing all the global variables.

Parameters: ξ_{lim} (Limit error)

Compute of the elastic solution $\mathcal{S}_0$ in dynamics.

$n = 1$ (Index of the LATIN iteration to be calculated)

while $\xi \geq \xi_{lim}$ **do**

- *Local stage* (Nonlinear constitutive relations):

Calculate $\rightarrow \hat{\mathcal{S}}_{n+1/2} \in \boldsymbol{\Gamma}$

- *Global stage* (Admissibility conditions):

Calculate $\rightarrow \mathcal{S}_{n+1} \in \boldsymbol{A_d}$

- LATIN Error calculation ξ_{n+1}.

$n = n + 1$

end

$\bar{\mathcal{S}} = \mathcal{S}_n$

4.6 Numerical examples

This section details numerical implementations of the LATIN-PGD method in dynamics. These tests consist of a 3D bending beam in dynamics subjected to a displacement in the "z" direction on both sides of the beam and corresponding zero displacements in the other directions as illustrated on figure 4.13.

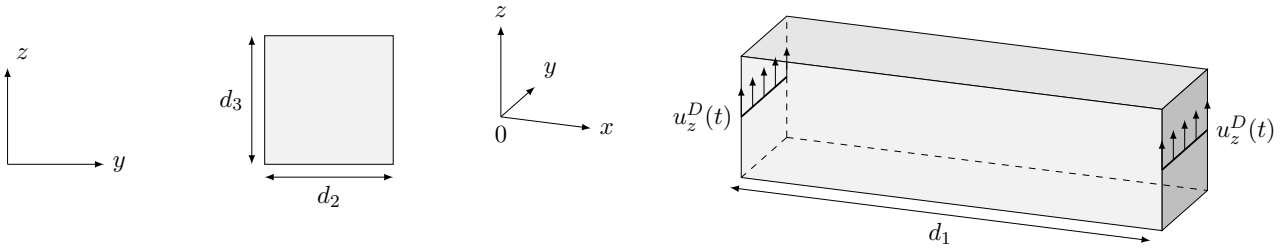


Figure 4.13: Test case considered, along with its dimensions.

The numerical tests that will be introduced in the following have the objective of illustrating the capabilities of the LATIN-PGD methodology. These tests are academic due to their relatively low number of degrees of freedom; intensive tests are left for future works. In addition, with the aim of presenting the computational time-savings of the LATIN method, the results will be compared against a classical step-by-step resolution, where Newmark scheme is employed for time-integration and a Newton-Raphson scheme for handling the material nonlinearity. More precisely, the equilibrium residual is minimized using a quasi-Newton strategy where a preliminary Cholesky decomposition of the associated operator to invert is performed in order to compare the performances of the proposed developments with a classical but robust and powerful

algorithm (see appendix B). To measure the error between the LATIN-PGD and Newton-Raphson solution the following error indicator is defined:

$$\xi = 100 \sqrt{\left(\frac{\left|\left|\left| \underline{\underline{\sigma}}_{\text{NR}} - \underline{\underline{\sigma}}_{\text{LATIN}} \right|\right|\right|^2}{\left|\left|\left| \underline{\underline{\sigma}}_{\text{NR}} \right|\right|\right|^2} + \frac{\left|\left|\left| \underline{\underline{\varepsilon}}_{\text{NR}} - \underline{\underline{\varepsilon}}_{\text{LATIN}} \right|\right|\right|^2}{\left|\left|\left| \underline{\underline{\varepsilon}}_{\text{NR}} \right|\right|\right|^2} \right)} \, [\%] \tag{4.169}$$

with $\left|\left|\left| \underline{\underline{\bullet}} \right|\right|\right|^2 = \int_{\Omega \times I} (\underline{\underline{\bullet}}) : (\underline{\underline{\bullet}}) \, d\Omega dt$.

The variables $\underline{\underline{\sigma}}_{\text{NR}}$, $\underline{\underline{\sigma}}_{\text{LATIN}}$, $\underline{\underline{\varepsilon}}_{\text{NR}}$ and $\underline{\underline{\varepsilon}}_{\text{LATIN}}$ correspond to the stress and total deformation tensors obtained by the Newton-Raphson (NR) and LATIN solver. It should be noted that both solvers use different numerical strategies for the simulation of their time response. On the one hand, the NR method is chosen to use the Newmark method due to its incremental resolution strategy in time, while on the other hand the LATIN-PGD use the Time Discontinuous Galerkin Method. To simplify the resolution of the problem, the time domain is uniformly discretized, however the difference in the time scheme considered implies a different number of temporal DOFs to be solved for both solvers, in this sense, the table 4.2 shows the number of DOFs in time for both solvers (n_T) as a function of the total number of temporal discretized elements (N_T).

Solver	Temporal DOFs (n_T)
NR	$2N_T + 1$
LATIN-PGD	$4N_T$

Table 4.2: Temporal DOFs for each solver.

The number of DOFs for the NR case is given by the simple choice that on each temporal element there exist 3 nodal values that must be determined, this is chosen in order to ensure a smooth temporal response. On the other hand, the LATIN-PGD with the TDGM needs the definition of 4 unknowns by temporal element, due to its Discontinuous approach and use of cubic Hermite polynomials (see section 2.2.3.2).

In the following, two numerical tests will be presented, where each one considers a different constitutive relation, being isotropic damage for concrete material and elasto-visco-plasticity for steel. The LATIN threshold considered for convergence are chosen differently for the two materials. This is because numerical tests have shown that isotropic damage requires a lower indicator than elasto-visco-plasticity to produce accurate results compared to the incremental nonlinear NR solver. In this regard, to ensure good results, the table 4.3 summarizes the LATIN and NR error thresholds considered for each case:

Constitutive relation	LATIN error threshold $[\%]$	NR error (norm of the equilibrium)
Isotropic damage	0.1	10^{-4}
Elasto-visco-plasticity	1	10^{-4}

Table 4.3: LATIN error threshold considered for each material behavior.

In contrast to the LATIN error, which measures the distance between the local and global solutions manifolds, the error threshold of the Newton-Raphson method is determined by computing a norm over the equilibrium equation (see appendix B). Both solvers use different error definitions, which makes it difficult to compare them under equal conditions. In this context, the error thresholds considered for both solvers were chosen based on empirical results only. Further research should be conducted to relate both errors in order to compare both solvers under equivalent conditions.

4.6.1 Isotropic damage example

In the present subsection, the LATIN method is considered for solving an isotropic damage constitutive relation in dynamics. For this test we consider the following dimensions for the beam presented in figure 4.13.

$$\begin{aligned} d_1 &= 8 \text{ [m]} \\ d_2 &= d_3 = 0.3 \text{ [m]} \end{aligned} \tag{4.170}$$

The prescribed vertical displacement $u_z^D(t)$ is shown in figure 4.14, which consists of a sinusoidal excitation of frequency 3 [Hz].

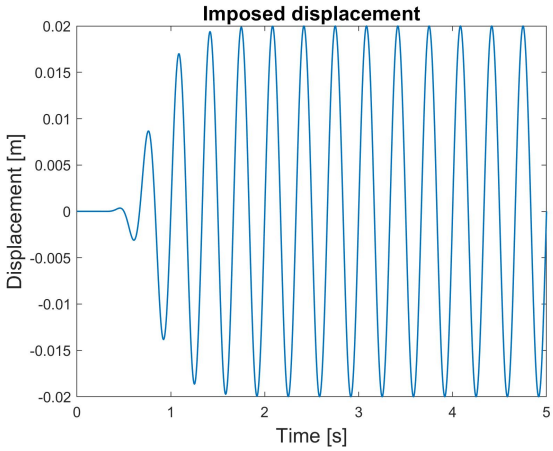
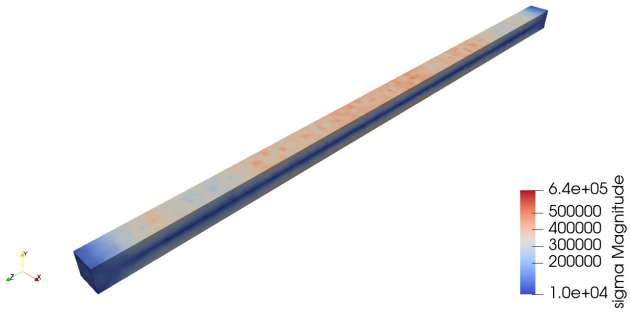
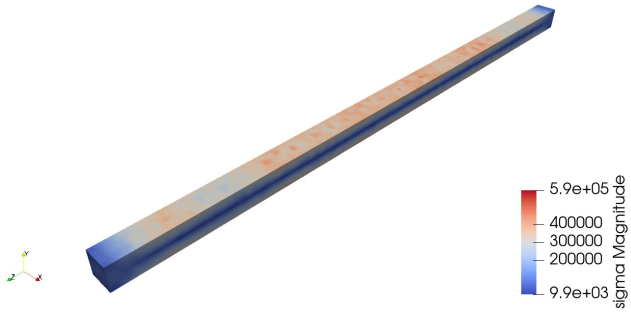


Figure 4.14: Imposed displacement.

Under this condition the magnitude of the stress tensor field on the beam at the end of the time interval ($t = T$) given for the LATIN-PGD and the NR method is presented in figure 4.15.



(a) Stress magnitude field obtained by the LATIN-PGD.



(b) Stress magnitude field obtained by the NR.

Figure 4.15: Magnitude of the stress tensor obtained with the LATIN-PGD (a) and NR (b) at the end of the temporal domain ($t = T$) for the case of isotropic damage.

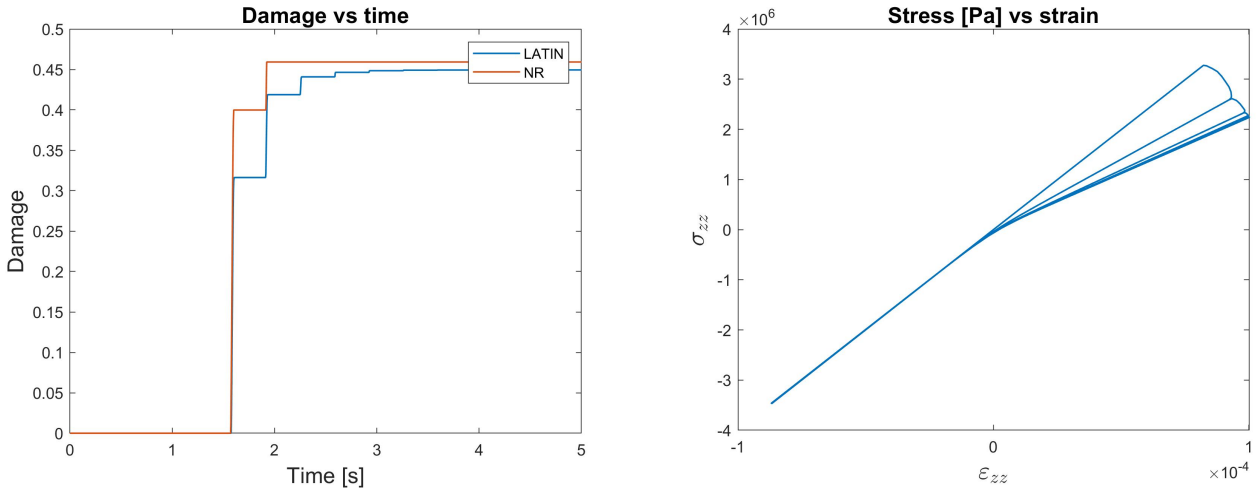
In figure 4.15 a difference in the maximum value obtained for the stress magnitude for both solvers can be observed. This difference in the solution of both solvers can be also seen in the graph of the evolution of the damage variable over the most requested integration point in space, given in figure 4.16a. From this figure we can see that there is a clear difference between the damage evolution obtained by the global LATIN-PGD and the incremental NR along the time domain. This difference is due, on the one hand, to the

different approximation strategy considered for the simulation of the time response and, on the other hand, to the inherent approximation of the solution delivered by the LATIN-PGD method, where the elastic and the LATIN corrections are approximated by a low-rank decomposition. In spite of the above, the maximum damage level of the structure is well reproduced, these values are shown in the table 4.4.

Solver	Damage
NR	0.4592
LATIN-PGD	0.4493

Table 4.4: Comparison of Damage.

Figure 4.16b show the stress vs. strain curve on the most solicited integration point in space, from this curve it can be seen that the phenomena of unilateral effect present in concrete materials is well reproduced.



(a) (b)

Figure 4.16: (a) Damage comparison between the LATIN-PGD and the NR method; (b) Stress vs strain curve on the most solicited integration point by using the LATIN-PGD method.

As mentioned previously, the time evolution technique considered for both solvers is completely different, while the Newton-Raphson solves the problem on the nodal values arising from a uniform discretization in time, the LATIN-PGD solves it on the nodes related to the FEM formulation using TDGM. In this sense, the total DOFs related to the reference problem, which are the multiplication between the space and temporal ones are resumed in table 4.5 for each solver.

Solver	Space DOFs	Time DOFs	Total DOFs
NR	6,471	901	5,830,371
LATIN-PGD	6,471	1,800	11,647,800

Table 4.5: DOFs of the reference problem.

Under these considerations, the resolution time obtained for each solver is resumed in table 4.6.

Solver	Computing time
NR	10.5 minutes
LATIN-PGD	4.3 minutes

Table 4.6: Time comparison for the case of isotropic damage.

From the above results we can highlight a 59 [%] computational gain enabled by the LATIN-PGD approach when compared to classical incremental solver, even if the total DOFs in time doubled. The graph of the LATIN error vs the number of modes computed for the reference problem is shown in figure 4.17.

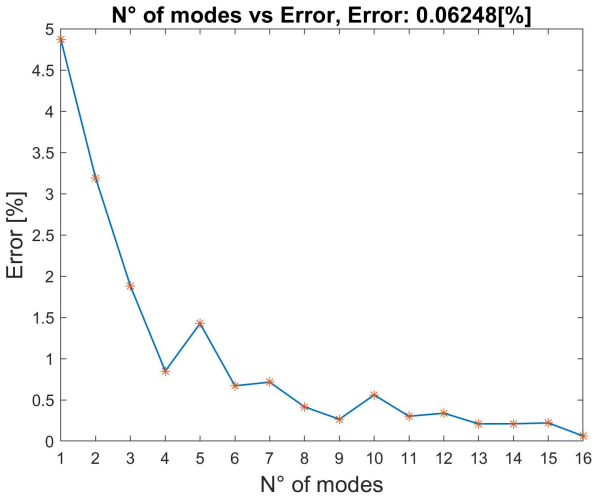


Figure 4.17: LATIN error vs PGD modes for the isotropic damage case.

For this particular problem, the required number of space-time PGD functions needed to converge is 16. This low number of modes gives the LATIN-PGD method its superior performance against the incremental solver. Finally, by using the error indicator (4.169) between the LATIN-PGD and the Newton-Raphson we obtained a value of 1.7497 [%].

4.6.2 Elasto-visco-plasticity example

Let us now consider a steel material with an elasto-visco-plasticity behavior. The dimensions of the considered beam are different from the case of concrete of section 4.6.1; these are chosen as:

$$\begin{aligned} d_1 &= 6 \text{ [m]} \\ d_2 = d_3 &= 0.1 \text{ [m]} \end{aligned} \tag{4.171}$$

In addition, the imposed vertical displacement $u_z^D(t)$ considered is shown in figure 4.18, where the signal frequency is also 3 [Hz] but of different amplitude.

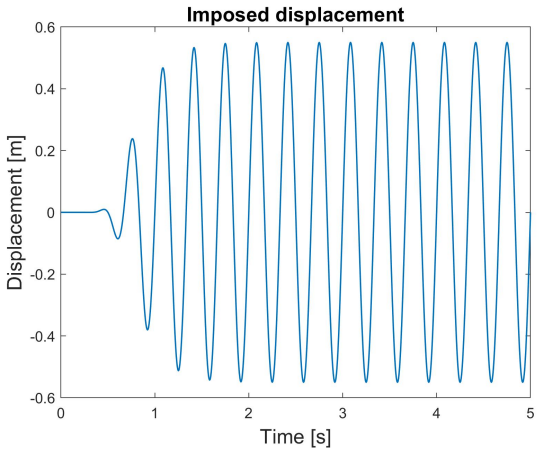


Figure 4.18: Imposed displacement.

Under this considerations the beam undergoes plasticity due to its inertia. Figures 4.19 and 4.20 shows the stress magnitude on the beam at the last time step obtained using the LATIN-PGD and the NR respectively.

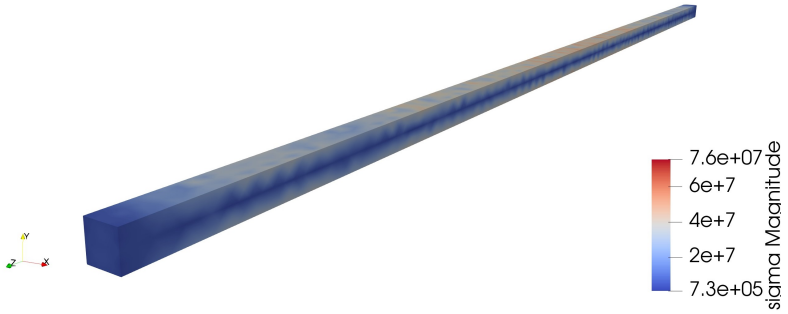


Figure 4.19: Stress magnitude field obtained by the LATIN-PGD at ($t = T$).

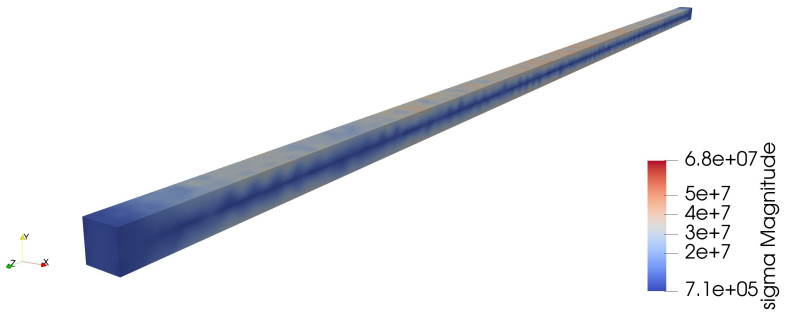
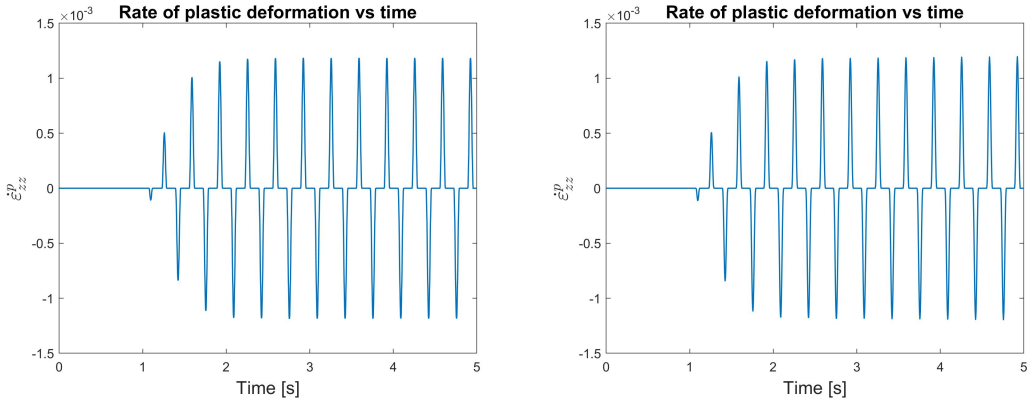


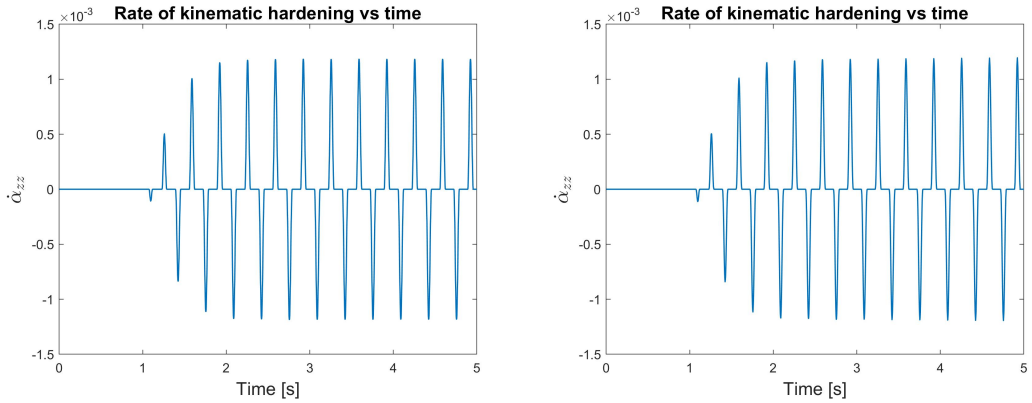
Figure 4.20: Stress magnitude field obtained by the NR at ($t = T$).

In order to compare the degree of accuracy of the LATIN-PGD, the plots of the rate of: plastic deformation, kinematic and isotropic hardening obtained from the most solicited integration point are shown in figures 4.21, 4.22 and 4.23 respectively, comparing the results obtained using the LATIN-PGD and the NR method.



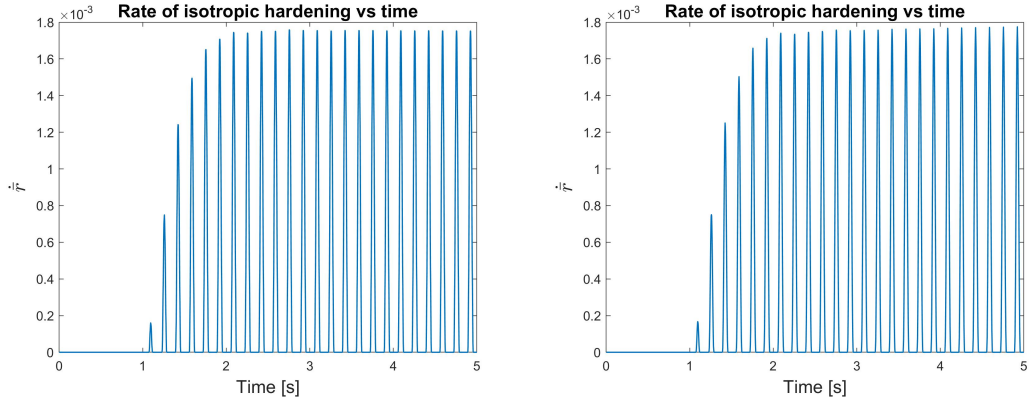
(a) (b)

Figure 4.21: Rate of plastic deformation of LATIN-PGD (a) and NR (b) on the most solicited integration point.



(a) (b)

Figure 4.22: Rate of kinematic hardening of LATIN-PGD (a) and NR (b) on the most solicited integration point.



(a) (b)

Figure 4.23: Rate of isotropic hardening of LATIN-PGD (a) and NR (b) on the most solicited integration point.

The present test have a different spatial discretization compared to the concrete material test; this is mainly due to the different spatial dimensions considered, however the temporal domain is discretized in the same way. The DOFs related to the space and temporal domain of the reference problem are given in table 4.7.

Solver	Space DOFs	Time DOFs	Total DOFs
NR	7,239	901	6,522,339
LATIN-PGD	7,239	1,800	13,030,200

Table 4.7: DOFs of the reference problem.

The computational time required for both solvers for the resolution of the problem are resumed in table 4.8.

Solver	Computing time
NR	6.3 minutes
LATIN-PGD	2.8 minutes

Table 4.8: Time comparison for the case of elasto-visco-plasticity.

In this case the LATIN-PGD achieves a 55.55 [%] reduction in computational cost compared to the incremental solver. The graph of the LATIN error versus the number of PGD modes is shown in figure 4.24.

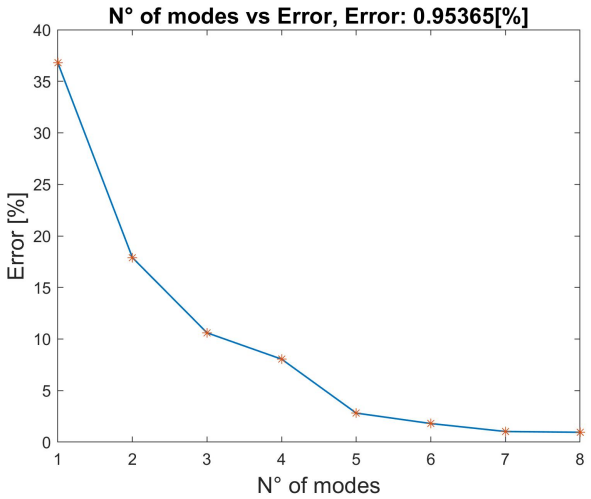


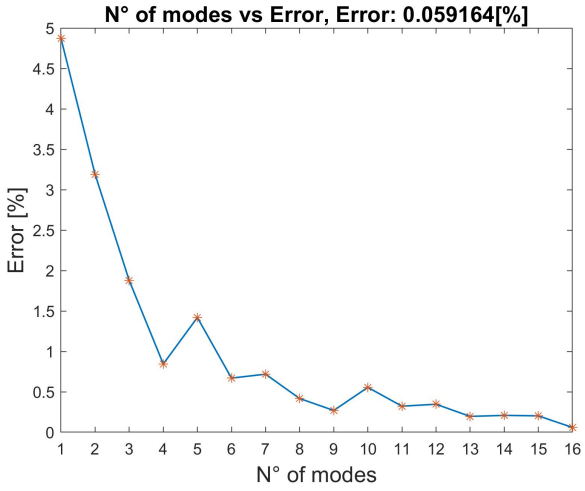
Figure 4.24: LATIN error vs PGD modes for elasto-visco-plasticity behavior.

Finally, the error between both methods following equation (4.169) gives a value close to 1 [%], confirming by consequence that the LATIN-PGD approach gives good results in low frequency dynamics problems when dealing with elasto-visco-plasticity.

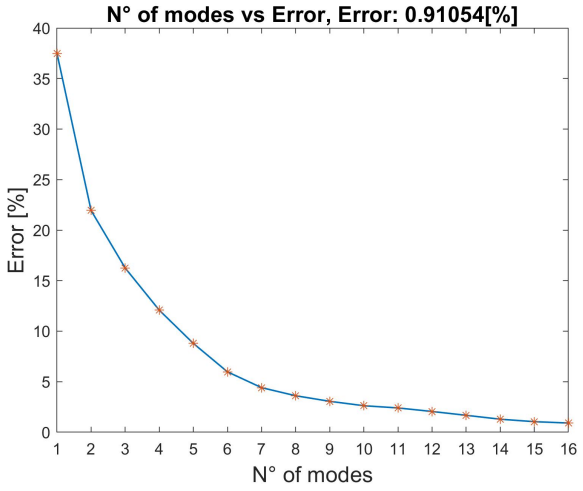
4.6.3 Comparison between the discontinuous (TDGM) and continuous (TCGM) approaches for the temporal resolution in the LATIN-PGD method

As shown in the present chapter, the Time Discontinuous Galerkin Method (TDGM) is used for the resolution of the PGD time functions in the LATIN method. This discontinuous formulation allows an incremental

resolution of the time functions while making use of a finite element formulation. As presented in previous sections, the incremental resolution of the temporal PGD functions in both the enrichment and preliminary steps, allows to optimize the temporal resolution when the temporal domain is relatively large, because only small size matrices (corresponding to the FEM formulation in a temporal element) must be inverted for its resolution. This is a considerable advantage over a continuous formulation using the Time Continuous Galerkin Method (TCGM), where the finite element formulation over the entire time interval $I = [0, T]$ requires the inversion of large matrices, both in the enrichment stage and in the preliminary stage, the latter being the most sensitive due to the determination of a set of time functions. To show the advantage of TDGM over TCGM, the numerical test considered in the previous section are solved by considering the same LATIN error thresholds given in table 4.3. The error curves versus the number of PGD modes obtained while solving the nonlinear problems using the TCGM are shown in figure 4.25, where the figures 4.25a and 4.25b shows respectively the results obtained for the cases of isotropic damage and elasto-visco-plasticity.



(a) Error vs PGD modes for the numerical test of section 4.6.1 considering isotropic damage and TCGM.



(b) Error vs PGD modes for the numerical test of section 4.6.2 considering elasto-visco-plasticity and TCGM.

Figure 4.25: LATIN error versus number of PGD modes for the case of isotropic damage and elasto-visco-plasticity considering a continuous formulation in time.

The following table 4.9 summarizes the computational times needed to converge for TCGM and recalls those needed for TDGM, which were obtained in sections 4.6.1 and 4.6.2 for the isotropic and elasto-visco-plasticity damage case, respectively.

Method	Isotropic damage	Elasto-visco-plasticity
TDGM	4.3 minutes	2.8 minutes
TCGM	5.8 minutes	7.9 minutes

Table 4.9: Time comparisons between the TDGM and the TCGM for the resolution of the numerical tests of sections 4.6.1 and 4.6.2.

From the table 4.9, we can see that the TDGM achieves a shorter computational time required for solving the nonlinear problems, in addition to a faster convergence rate, especially when elasto-visco-plasticity is involved. In fact, as observed in figure 4.25b, when using TCGM the total number of modes required to converge is 16, while for TDGM only 8 are required to reach the same LATIN threshold of $1[\%]$. This is a

very interesting property, which could be explained by recalling that the TDGM allows a small discontinuity between the temporal elements of the FEM formulation. This discontinuity slightly modifies the temporal PGD functions obtained, thus positively modifying the convergence rate of the LATIN method.

Remark: It is worth noting that the advantages of TDGM over TCGM for solving PGD time functions increase significantly when a longer time interval is considered.Thus, the difference in computational times shown in the table 4.9 are larger as the time interval considered is longer, positioning TDGM as a very attractive and efficient strategy for the fast resolution of nonlinear problems in the context of the LATIN-PGD.

4.6.4 Conclusions on the numerical results

From the numerical results shown in the lasts sections, we saw a gain in performance of the LATIN-PGD method compared to the classical incremental solver for both isotropic damage and elasto-visco-plasticity. This last is achieved due to the use of the model reduction technique PGD, which allows to accelerate the calculations. However, it should be noted that this reduction in computational time is due to the fact that the solution of the nonlinear problem can be approximated by a low-rank decomposition. If the nonlinear solution we are trying to obtain does not support a low-rank decomposition, the LATIN-PGD method is likely to decrease its performance and end up being more expensive than the incremental solver. This point is of main importance and is one of the reasons why the extension of the LATIN-PGD method is limited to low-frequency dynamics problems. This means that the problems that can be solved with good numerical performance are those that do not involve propagative solutions in the solid medium. The approximation of propagative solutions, i.e. where the spatial solution is completely coupled with its temporal response, does not allow a low-rank approximation, which would limit the effectiveness of the LATIN-PGD method for this type of problems. However, in the domain of the dynamics of massive structures, which is the area studied in the present thesis, the solution lies within the range of slow dynamics, where the present extension of the LATIN-PGD method can be applied without problems.

An additional point to note is the accuracy of the solution obtained. From the results sections, it is clear that the LATIN-PGD and the incremental solver give solutions that differ slightly from each other, especially the damage evolution in the case of isotropic damage as presented in figure 4.16. This is due to all the approximate considerations that help to speed up the resolution in the LATIN-PGD, starting from an approximate elastic solution employing the modal basis of the system up to the use of the PGD in the global stage of the LATIN method. All these points have an associated error that contributes to the difference found in the results. However, the error between the LATIN-PGD and the incremental solver remained below 2 [%] for both constitutive relations considered, showing that the solver allows to correctly approximate the final nonlinear solution.

Finally, as can be seen in section 4.6.3, the Time Discontinuous Galerkin Method (TDGM) used in this chapter demonstrated better performance when solving nonlinear problems compared to the continuous approach (TCGM) in the context of the LATIN-PGD (where the efficiency is higher when the time domain is large). The above is explained by the fact that when using the TDGM the temporal functions are determined incrementally in time (for each finite time element), which results in the inversion of small size matrices, allowing to optimize the temporal resolution in the enrichment and preliminary stages of the LATIN-PGD method. In addition, the convergence rate of the LATIN method is improved, allowing a faster convergence to the reference solution.

All the aforementioned points makes the LATIN-PGD method along with the TDGM an attractive solver when performance, accuracy and reliability are the main considerations when solving low-frequency dynamics problems.

4.7 Conclusions

In this chapter, the LATIN-PGD method is reformulated to introduce inertial effects when dealing with low-frequency dynamics nonlinear problems. This extension is presented for the case of two constitutive relations corresponding to isotropic damage in concrete and elasto-visco-plasticity for metals, but other types of materials could be considered without major difficulty. The LATIN-PGD allows to reduce the computational time in comparison with classical incremental solvers when solving with nonlinear problems. Such reduction is directly proportional to the number of degrees of freedom considered for both spatial and temporal domain. The LATIN is a non-incremental resolution method that enables the use of the model order reduction PGD to express the nonlinear solution of the problem as a low-rank approximation. The PGD allows to capture the redundancies of the nonlinear solution in both space and time, allowing a considerable simulation time reduction.

In addition, an incremental time resolution strategy based on the Time Discontinuous Galerkin Method (TDGM) was introduced, which allows to efficiently solve the temporal PGD functions, since it allows to reduce the size of the operators needed to be inverted for the enrichment and preliminary steps of the LATIN-PGD method. This is very useful and imposes an advantage over the classically used methods, which correspond to a continuous formulation in time using the Time Continuous Galerkin Method (TCGM), especially when the time domain is large. In these situations, the continuous formulation requires the assembly and inversion of large matrices for the temporal resolution, which decreases the efficiency of the LATIN-PGD method. On the other hand, in the case of the discontinuous formulation using the TDGM, the temporal PGD functions are computed element by element of the temporal FEM discretization, avoiding the construction and inversion of large assembled matrices, thus increasing the efficiency of the LATIN-PGD method.

Despite the performance of the LATIN-PGD method and the efficiency in time resolution that TDGM allows, there are still certain problems where these ideas are not enough to effectively reduce the computational cost of their resolution. Among them are the problems in which the input excitation has a long duration in time, for example, seismic excitations or fatigue signals of rich frequency content, which are typical excitations to which buildings are subjected and therefore their treatment is a fundamental part of this thesis work. The classical application of the LATIN-PGD to solve these types of problems can consume prohibitive computational resources due to the large number of degrees of freedom in time that must be determined in the global stage, as well as the increased cost related to the evaluation of the constitutive relations quantities in the local stage. The loss of efficiency of the LATIN method in this sense is mainly due to its inherent global solver nature, which implies the manipulation of large size matrices for its correct application, reducing in this sense its performance.

In this context, in the next chapter, a new multiscale temporal approximation is presented, which aims to optimize the resolution of the PGD temporal functions when excitations of long duration and complex behavior are considered. Additionally, due to the high computational cost related to the integration of the constitutive relation in the whole spatio-temporal domain when the temporal domain is large (spatial as well), a hyper-reduction technique applied to the local stage is introduced in chapter 6. This technique allows not only to speed up the evaluation of the local stage variables, but also to express them as a low-rank approximation that reduces memory consumption and optimizes the integration operations of the global stage, thus achieving a more optimized LATIN-PGD method when dealing with problems of large temporal duration.

Chapter 5

LATIN-PGD multiscale in time for the resolution of complex fatigue problems

As seen in the previous chapters, the LATIN-PGD method allows to decrease the computational cost when solving nonlinear problems. This increase in performance is mainly due to the use of a low-rank PGD approximation in the global stage, which allows to reduce a large coupled space-time problem to a smaller one decoupled in space and time. However, there are problems in which this reduction in computational cost is not sufficient for the rapid resolution of the problem, for example when considering external excitation of long duration and rich frequency content, such as seismic loads or complex fatigue inputs. In this case, one of the main factors that limit the performance of the solver is the determination of a large number of degrees of freedom (DOF) of the discretized temporal problem at each global stage. In order to overcome this limitation, a new multiscale approximation in time is proposed in this chapter. This consists of (i) a method dedicated to the multiscale approximation of the external input excitation and (ii) a multiscale approximation of the PGD temporal functions in the LATIN-PGD method. This new multiscale strategy aims not only at optimizing the LATIN-PGD solver, but also seeks to provide new tools and a framework that could help in the future to introduce new ideas to further optimize the LATIN-PGD method when solving nonlinear problems considering long duration excitations with rich frequency content. To illustrate the method, the chapter ends with numerical examples considering isotropic damage in concrete and elasto-visco-plasticity in metals.

Contents

5.1 Introduction . . . 100
5.2 Novel temporal multiscale approximation of signals . . . 107
5.3 New temporal multiscale approach applied to the LATIN-PGD solver . . . 114
5.4 Numerical examples . . . 129
5.5 Conclusions . . . 140

5.1 Introduction

As seen in chapter 4, the use of the PGD into the LATIN formulation allows to increase the computational efficiency of the solver. However, despite this advantage, if external excitations are of long duration as could be the case of seismic signals or fatigue loads, computing the LATIN-PGD solution could become really expensive and prohibitive. This increase in computational cost is due to the large number of DOFs in time corresponding to the discretization of the temporal functions in the large time domain (when using the Time Discontinuous Galerkin method or a continuous approach as the Time Continuous Galerkin method). For instance, if we consider a monoperiodic excitation composed of 10^5 cycles and a discretization such that within each cycle there are 10 time-elements, the total number of DOFs to be determined is in the order of one million, which is an exorbitant amount and a real limitation if such problem need to be solved fast.

For particular cases where the external excitation (imposed displacement, external forces, etc.) has a unique periodic pattern that repeats throughout the time domain and whose amplitude changes slowly, it is possible to apply the classical multiscale approximation introduced in section 3.4.3. This technique assumes that the nonlinear evolution response does not undergo abrupt changes over the time interval so that the time response can be easily interpolated. This technique consists in applying an interpolation of the response in the time domain knowing the solution over so-called nodal cycles that are defined in specific time subdomains distributed along the entire time interval.

However, if the external loading has a richer frequency content and a long duration in time, there exists no dedicated strategies for its approximation. In this situation only a classical LATIN-PGD resolution such as the one presented in chapter 4 can be used for the temporal resolution, resulting in increased computational costs. This imposes a serious drawback, especially when a temporal problem has to be solved several times within the global stage on the *enrichment* and *preliminary* steps, as well as when parametric studies have to be performed.

To overcome this limitation, the present chapter introduces a new multiscale strategy, capable of macroscopically interpolating a complex multiperiodic solution. This method is based on a new micro-macro formulation, which is composed of two main ingredients: (i) the multiscale approximation of the external load and (ii) the multiscale approximation of the temporal PGD functions in the LATIN solver.

The new multiscale strategy presented in this chapter aims to provide new tools in the signal theory domain to optimize the resolution of nonlinear problems when the external load has complex behavior and long time duration.

To properly understand the main ideas of this new approach, it is first necessary to give a quick overview of the previous multiscale versions of the LATIN-PGD, in order to highlight some ideas that will be of key importance for the development of the new strategy. The previous multiscale versions and the new strategy are introduced in the following sections.

5.1.1 Previous multiscale approximation for simple fatigue problems

The first work that tried to exploit a multiscale strategy for the approximation of the time evolution in the LATIN method was presented in [Cognard and Ladevèze, 1993] where a strategy for solving cyclic fatigue problems at low numerical cost in the framework of elasto-visco-plasticity was described. The resolution of the problem on the full interval $I = [0, T]$ is replaced by an approximated resolution on a coarse regularly-spaced discretization called *macro intervals*. Each macro interval is limited at the beginning and at the end by the so called *nodal cycles* as shown in figure 5.1. From the solution known on the nodal cycles, the solution on the full interval $I = [0, T]$ is interpolated by using specific shape functions (linear or quadratics depending on the external excitation's signal [Cognard and Ladevèze, 1993] or [Arzt and Ladevèze, 1994]

respectively) that evolve slowly in time. The determination of the solution in the nodal cycles (without computing the solution in-between them), is performed by estimating an initial condition in each nodal cycle and interpolating the operator used to solve the temporal problem in the global stage, both under the hypothesis of a slow macro evolution of the temporal solution (see [Ladevèze, 1999]).

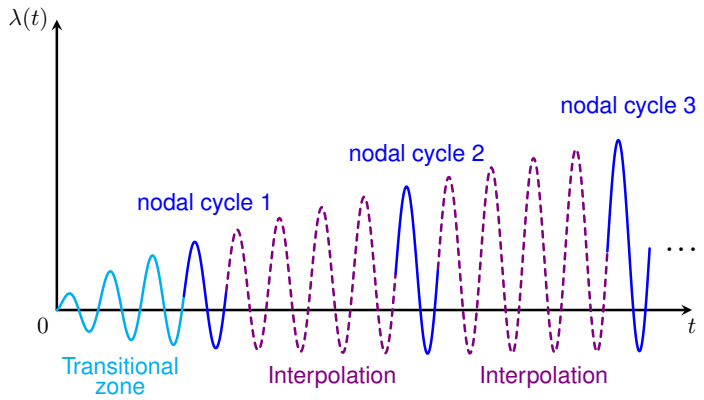


Figure 5.1: Cyclic approximation – definition of nodal cycles.

The strategy proposed in [Cognard and Ladevèze, 1993] is well suited for sinusoidal fatigue loading but needs to be extended for more complex inputs. Indeed, the main hypothesis for the application of this multiscale approach lies in the periodicity and slow evolution of the excitation amplitude. This is not verified if a multi-frequency signal is considered, for example the one presented in figure 5.2. This signal presents such a complicated behavior that makes a multiscale approximation based on the interpolation of nodal cycles impossible.

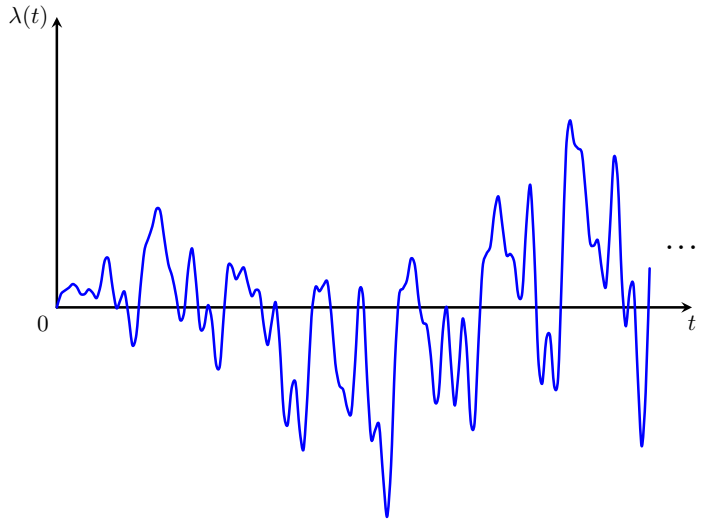


Figure 5.2: Complex signal where the previous temporal multiscale approximation can't be applicable.

For complex and long time duration external excitations, the classical multiscale approximation of the LATIN-PGD method is no longer applicable, therefore, for the multiscale approximation of such problems it is necessary to consider a more general approach.

5.1.2 New temporal multiscale approximation

As already mentioned, for the case of complex excitations with long time duration, the previous multiscale strategy is no longer applicable. Therefore, in this section a new temporal multiscale approximation is presented [Ladevèze, 2018]. The main idea consists in approximating the temporal PGD basis functions as the sum of multiscale *sub-modes*, under the form:

$$\lambda(t) \approx \sum_{\ell=1}^{m_s} \bar{\lambda}_\ell(t;\tau_\ell) = \sum_{\ell=1}^{m_s} A_\ell^{\mathsf{I}}(t) h_\ell^{\mathsf{I}}(t;\tau_\ell) + A_\ell^{\mathsf{R}}(t) h_\ell^{\mathsf{R}}(t;\tau_\ell) + A_\ell^{o}(t) \tag{5.1}$$

Each sub-mode $\bar{\lambda}_\ell$ is composed of macro functions $A_\ell^{\mathsf{I}}(t)$, $A_\ell^{\mathsf{R}}(t)$, $A_\ell^{o}(t)$ and micro functions $h_\ell^{\mathsf{I}}(t)$ and $h_\ell^{\mathsf{R}}(t)$. The micro functions correspond to patterns that quickly repeat over time and the macro terms correspond to the envelope of the micro functions ($A_\ell^{\mathsf{I}}(t)$ and $A_\ell^{\mathsf{R}}(t)$) plus an additional free term ($A_\ell^{o}(t)$), whose evolution is slow over time. The micro functions are separated into two groups, the symmetric ($h_\ell^{\mathsf{R}}(t)$) and the antisymmetric ones ($h_\ell^{\mathsf{I}}(t)$). As an example lets consider the signal presented in figure 5.2, and lets apply the approximation of equation (5.1), in this way the function is simply approached by the sum of two multiscales sub-modes $(\bar{\lambda}_\ell)_{\ell=1}^{2}$, where the first one is given in figure 5.3.

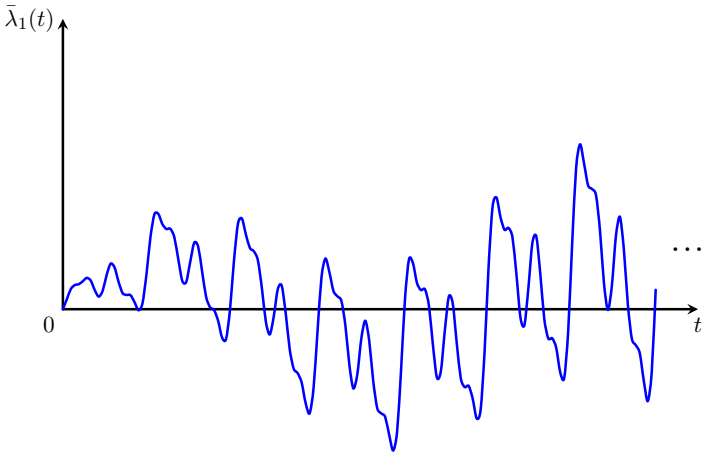


Figure 5.3: First temporal sub-mode.

While the second temporal sub-mode is given in figure 5.4.

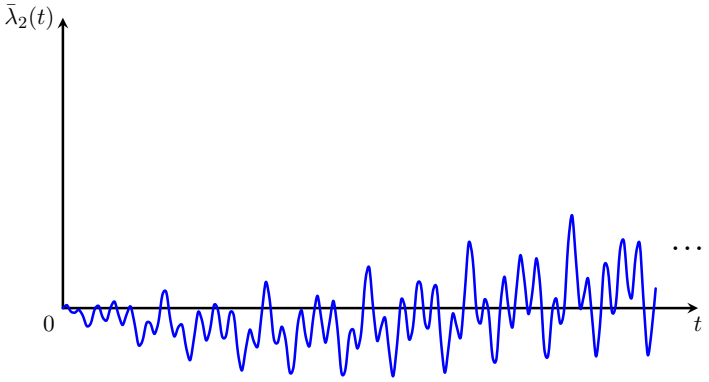
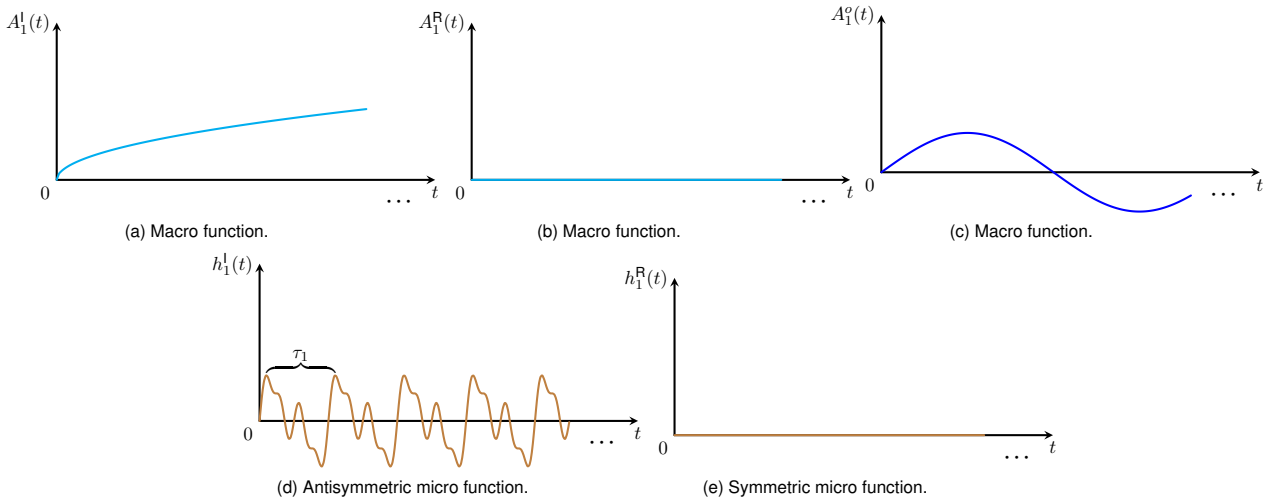


Figure 5.4: Second temporal sub-mode.

As already mentioned, the micro functions of each sub-mode consist of patterns that repeat over the

whole time interval at a given ***characteristic period*** denoted τ_ℓ as can be seen in figures 5.5 and 5.6, where the macro and micro functions are also illustrated.

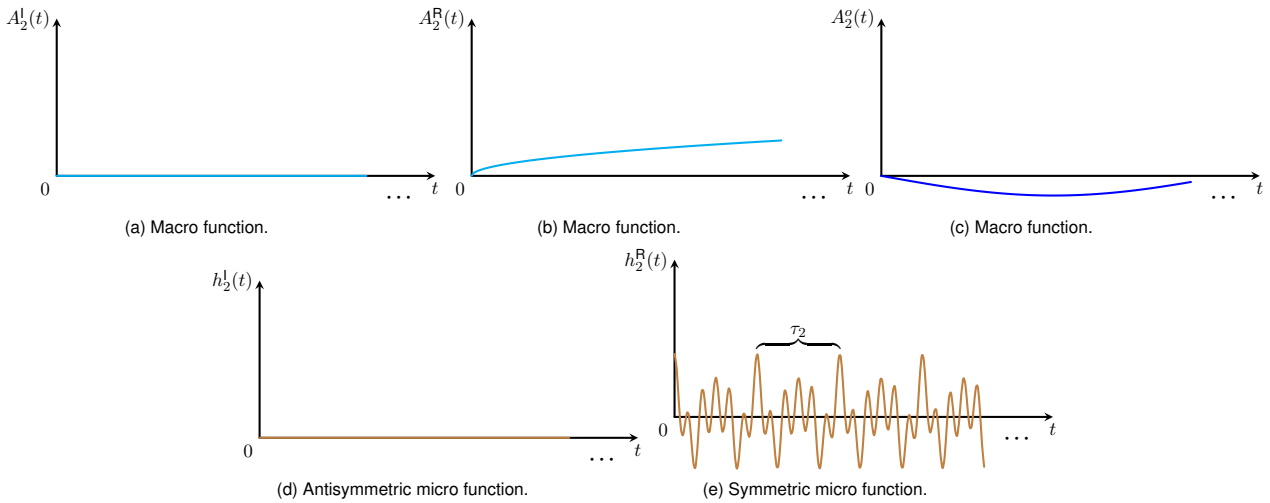


(a) Macro function. (b) Macro function. (c) Macro function.

(d) Antisymmetric micro function. (e) Symmetric micro function.

Figure 5.5: Macro and micro functions of the first sub-mode.

The determination of the characteristic periods of each sub-mode is performed from a frequency study of the external excitation applied to the structure. This idea can be visualized in a simple way by considering the case of a monoperiodic external excitation, where the characteristic period is well known *a priori* (excitation period), in this particular case it is known that all PGD time functions will present the same characteristic period along their evolution. The determination of this important period for the case of a monoperiodic excitation is straightforward and simple, however, if the external excitation corresponds to a seismic signal or some fatigue loading with a complex behavior in time, the determination of these characteristic periods becomes complicated. To this end, we introduce a novel method in section 5.2 that achieves this requirement and approximates the input excitation as the sum of sinusoidal contributions. Once the characteristic periods of the external excitation are determined, they are used in the new multiscale approximation of the PGD time functions for the determination of the micro functions $h_\ell^{\mathsf{R}}(t)$ and $h_\ell^{\mathsf{I}}(t)$.



(a) Macro function. (b) Macro function. (c) Macro function.

(d) Antisymmetric micro function. (e) Symmetric micro function.

Figure 5.6: Macro and micro functions of the second sub-mode.

For the optimal construction of each temporal sub-mode of the equation (5.1) it is necessary to introduce some important considerations for the determination of the macro and micro functions, which are discussed below.

1. **Multiscale behavior of the temporal sub-modes**:

   In order for all sub-modes to have a multiscale behavior, the discretization of the macro functions must be such that within a finite macro element of size ΔT_ℓ there are several cycles of characteristic period τ_ℓ, this requirement can be expressed as:

   $\forall \ell \in [1, ..., m_s]$,

   $$\frac{\Delta T_\ell}{\tau_\ell} = n_c \quad , \quad n_c \in \mathbb{N} \tag{5.2}$$

   This condition imposes that within a macro finite element there are n_c cycles of characteristic period $(\tau_\ell)_{\ell=1}^{m_s}$ as illustrated in figure 5.7. The value n_c is constant and fixed for all the sub-modes. This condition is necessary to reduce the computational time associated with the determination of the macro functions, where the higher the constant n_c the fewer temporal DOFs need to be computed for each sub-mode ℓ.

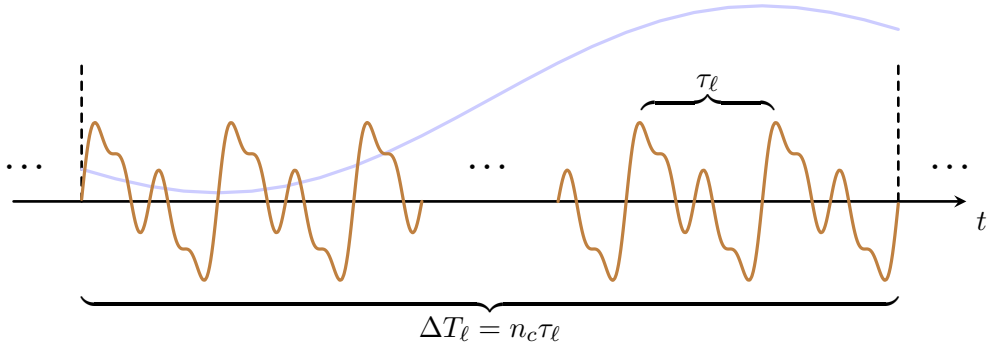


Figure 5.7: Macro and micro functions representation.

Condition (5.2) also implies that the different temporal sub-modes may have different macro discretizations, this particularity makes impossible the application of direct or update techniques as presented in chapter 3 for their optimal determination. By consequence, the above statements imply that the sub-modes must be determined incrementally by a “greedy” algorithm (see section 3.2.5.4).

The difference in the temporal sizes related to the micro and macro functions for each sub-mode requires a dedicated definition for each temporal domain, in this sense, as it will be widely used in the next sections, we introduce the temporal domain related to the macro functions as:

$\forall k \in [1, ..., N_\ell]$, $\forall \ell \in [1, ..., m_s]$,

$$I_{k,\ell}^M = [(k-1)\Delta T_\ell, (k)\Delta T_\ell] \tag{5.3}$$

where M stands for “macro” (to differentiate with the micro time interval defined below) and N_ℓ the total number of temporal FEM elements in the macro discretization for a given sub-mode ℓ. On the other hand, the domain related to the micro discretization is defined as:

$\forall p \in [1, ..., N_{c,\ell}]$, $\forall \ell \in [1, ..., m_s]$,

$$I_{p,\ell} = [(p-1)\tau_\ell, (p)\tau_\ell] \tag{5.4}$$

where we denote the total number of micro intervals equals to $N_{c,\ell} = n_c N_\ell$ for a given sub-mode ℓ.

From the above definitions, in order to keep the notations used in chapter 4, the index "k" denotes the macro element of the FEM discretization, while the index "p" denotes a micro interval. This convention will be used throughout this chapter.

2. **Continuity of the micro functions**:

   To guarantee the continuity of the sub-modes over the time interval, the micro functions must verify certain conditions. For instance, the antisymmetric functions must have a pattern with an initial and final value equal to zero, that is:

   $\forall p \in [1, ..., N_{c,\ell}]$, $\forall \ell \in [1, ..., m_s]$,

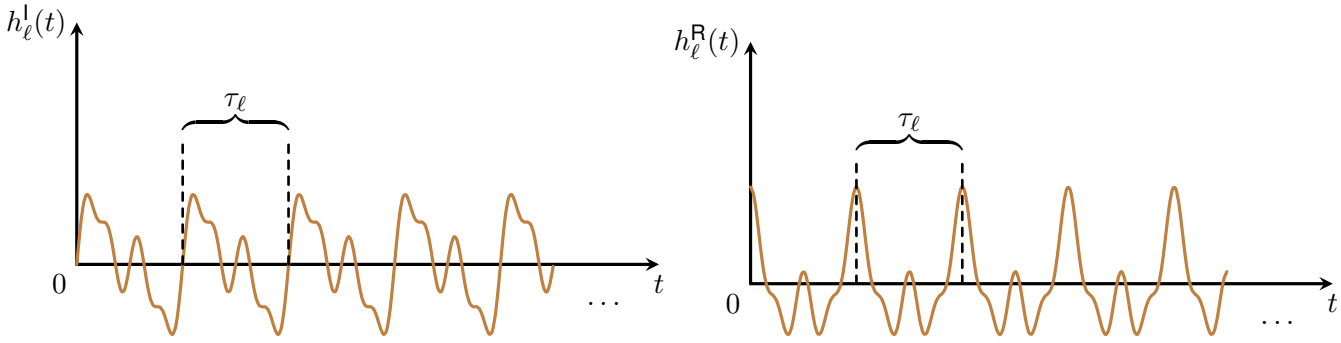
$$h_\ell^{\mathsf{I}}\Big|_{t=(p-1)\tau_\ell} = h_\ell^{\mathsf{I}}\Big|_{t=(p)\tau_\ell} = 0 \tag{5.5}$$

   On the other hand, symmetric micro functions simply verify the following condition:

   $\forall p \in [1, ..., N_{c,\ell}]$, $\forall \ell \in [1, ..., m_s]$,

$$h_\ell^{\mathsf{R}}\Big|_{t=(p-1)\tau_\ell} = h_\ell^{\mathsf{R}}\Big|_{t=(p)\tau_\ell} \tag{5.6}$$

   The micro functions presented in figures 5.8a and 5.8b illustrate these continuity conditions.



(a) Micro antisymmetric component of the signal.

(b) Micro symmetric component of the signal.

Figure 5.8: Micro functions of the temporal PGD multiscale approximation associated to the LATIN method.

3. **Approximation of the macro functions**:

   The approximation of the macro functions is performed by using the finite element method in time; the used shape functions correspond to the cubic Hermite polynomials, the approximation generated by these functions and their time-derivatives are continuous. The shape functions defined on a macro element for each of the sub-modes are given by:

   $\forall t \in I_{k,\ell}^M$, $\forall k \in [1, ..., N_\ell]$, $\forall \ell \in [1, ..., m_s]$,

$$\begin{aligned}
\Psi_{1,\ell}^{[k]}(t) &= \left(1 - \frac{3}{\Delta T_\ell^2}(t - t_{k-1})^2 + \frac{2}{\Delta T_\ell^3}(t - t_{k-1})^3\right) \\
\Psi_{2,\ell}^{[k]}(t) &= \left((t - t_{k-1}) - \frac{2}{\Delta T_\ell}(t - t_{k-1})^2 + \frac{1}{\Delta T_\ell^2}(t - t_{k-1})^3\right) \\
\Psi_{3,\ell}^{[k]}(t) &= \left(\frac{3}{\Delta T_\ell^2}(t - t_{k-1})^2 - \frac{2}{\Delta T_\ell^3}(t - t_{k-1})^3\right) \\
\Psi_{4,\ell}^{[k]}(t) &= \left(-\frac{1}{\Delta T_\ell}(t - t_{k-1})^2 + \frac{1}{\Delta T_\ell^2}(t - t_{k-1})^3\right)
\end{aligned} \tag{5.7}$$

where we considered $t_{k-1} = (k-1)\Delta T_\ell$, and the macro temporal domain $I^M_{k,\ell}$ defined as presented in equation (5.3).

Lets therefore define the vector of macro shape functions on a given element "k" of the macro discretization when computing the sub-mode "ℓ" as:

$\forall t \in I^M_{k,\ell}$, $\forall k \in [1, ..., N_\ell]$, $\forall \ell \in [1, ..., m_s]$,

$$\underline{\Psi}^{[k]}_\ell(t) = \left[\Psi^{[k]}_{1,\ell}(t), \Psi^{[k]}_{2,\ell}(t), \Psi^{[k]}_{3,\ell}(t), \Psi^{[k]}_{4,\ell}(t)\right]^T \tag{5.8}$$

Contrary to chapter 4 where the Time Discontinuous Galerkin Method is used, here a classic Continuous Galerkin approach is employed. This is considered due to the complexity of the multiscale formulation and to improve the continuity of the approximation, however the introduction of a discontinuous approach is proposed as a perspective for future works.

5.1.3 Important remarks

The previous temporal multiscale approach introduced in section 5.1.1 and the new multiscale technique have different considerations that must be highlighted. These considerations include two main points, which are the elastic solution and the local stage of the LATIN method, these are presented in the following subsections.

5.1.3.1 Initial elastic solution

The multiscale approximation presented in [Cognard and Ladevèze, 1993] was developed to the resolution of quasi-static problems and monoperiodic excitations, those considerations allowed the nodal cycle interpolation to be applicable also to the determination of the elastic solution.

The new multiscale strategy, on the other hand, can also be used for the determination of quasi-static elastic solutions, however, this is not the case of the present thesis work, where the elastic solution is determined under dynamics conditions. Dynamics problems introduced an additional difficulty, which corresponds to the inertial effects, these effects can produced transitional solutions in time which are complicated to approximate using a multiscale approach. For this reason, in the present chapter the elastic solution in dynamics is simple calculated over the whole time domain by using the Time Discontinuous Galerkin Method as presented in chapter 4. This discontinuous formulation allows solving long duration problems easily due to its incremental resolution in time, which positions TDGM as an efficient alternative for obtaining the elastic solution.

5.1.3.2 Local stage

When the external excitation is monoperiodic, the determination of the constitutive relation quantities at the local stage can also be interpolated by using the nodal cycles idea of the previous multiscale approach. However, this is not the case in the present chapter, where a complex external excitation is considered, since the solution at the local stage cannot be interpolated due to the complexity of the input signal coming from the global solution of the LATIN method. For this reason in this chapter the local stage is fully evaluated without any approximation. This can increase a lot the computational cost due to the whole evaluation of the constitutive relation on the large temporal domain. In order to decrease the computational efforts of the local stage evaluation, a hyper-reduction technique is introduced next in chapter 6, although this technique is not considered in this chapter in the numerical examples section.

5.2 Novel temporal multiscale approximation of signals

As presented in the introductory section, the new multiscale approximation needs the knowledge of the characteristic periods of the signal associated to the external excitation. To do so, here a multiscale approximation of signals initially described in [Ladevèze, 2018] is presented, this method allows in an easy and inexpensive way the determination of these characteristic periods.

To introduce the ideas, let us consider a given signal $s_{ref}(t)$, and decompose it by introducing two time scales, one micro and one macro. At the micro scale the evolution is cyclic and varies rapidly, however the evolution is slow at the macro scale. Under these considerations the signal is approximated as follows:

$$s_{ref}(t) \approx s_{m_s}(t) = \sum_{\ell=1}^{m_s} A_\ell^{\mathsf{R}}(t) \cos\left(\frac{2\pi t}{\tau_\ell}\right) + A_\ell^{\mathsf{I}}(t) \sin\left(\frac{2\pi t}{\tau_\ell}\right) + A_\ell^o(t) \tag{5.9}$$

where $s_{ref}(t)$ is the original signal that we want to approximate, and $s_{m_s}(t)$ its multiscale approximation considering m_s sub-modes.

In this expression, $A_\ell^{\mathsf{R}}(t)$, $A_\ell^{\mathsf{I}}(t)$ and $A_\ell^o(t)$ are the functions that represent the evolution at the macro scale of the approximation, they can be seen as the respective envelope of each sub-mode "ℓ" of the representation, "R" denotes the real part (cosine's envelope), "I" the imaginary part (sine's envelope) and $A_\ell^o(t)$ corresponds to the macro free term (a term that is not associated to a micro function). The micro evolution is given in a sinusoidal way which is function of their characteristic period τ_ℓ different for each sub-mode, which are the key variables of the method and the main unknowns needed to be determined. The procedure required to compute τ_ℓ and the macro functions will be explained below.

5.2.1 Determination of the multiscale sub-modes

The characteristic periods to be determined may have very different values from each other, which implies that a different macro discretization must be considered for each sub-mode. Therefore, each sub-mode must be determined incrementally or in a greedy manner. This incremental construction of the sub-modes is presented in detail in the following subsections.

5.2.1.1 "Greedy" calculation of multiscale sub-modes

In what follows, let's consider that we have already computed $\ell - 1$ sub-modes of the decomposition and that our objective is to determine the sub-mode "ℓ". Under these considerations we can define the following residual function:

$$r_\ell(t) = s_{ref}(t) - s_{\ell-1}(t) \tag{5.10}$$

where $s_{ref}(t)$ is the original signal that we want to approximate, and $s_{\ell-1}(t)$ the multiscale approximation given by:

$$s_{\ell-1}(t) = \sum_{i=1}^{\ell-1} \bar{s}_i(t) \tag{5.11}$$

where $\bar{s}_\ell(t)$ corresponds to the sub-mode ℓ of the signal decomposition, which is given as:

$$\bar{s}_\ell(t) = A_\ell^{\mathsf{R}}(t) \cos\left(\frac{2\pi t}{\tau_\ell}\right) + A_\ell^{\mathsf{I}}(t) \sin\left(\frac{2\pi t}{\tau_\ell}\right) + A_\ell^o(t) \tag{5.12}$$

The idea is to find the best sub-mode $\bar{s}_\ell(t)$ that minimizes the following functional J in the sense of a norm $L^2(0,T)$:

$$J = \|r_\ell - \bar{s}_\ell\|_I^2 = \int_0^T \left[r_\ell(t) - \left(A_\ell^{\mathsf{R}}(t) \cos\left(\tfrac{2\pi t}{\tau_\ell}\right) + A_\ell^{\mathsf{I}}(t) \sin\left(\tfrac{2\pi t}{\tau_\ell}\right) + A_\ell^o(t)\right)\right]^2 dt \tag{5.13}$$

By defining the vector of micro functions $\underline{n}_\ell$ and the vector of nodal macro amplitudes $\underline{\boldsymbol{A}}_\ell^{[k]}$ on a given element k by:

$$\underline{n}_\ell = \begin{bmatrix} \cos\left(\frac{2\pi t}{\tau_\ell}\right) \\ \sin\left(\frac{2\pi t}{\tau_\ell}\right) \\ 1 \end{bmatrix} \tag{5.14}$$

$$\underline{\boldsymbol{A}}_\ell^{[k]} = \begin{bmatrix} (\underline{\boldsymbol{A}}_\ell^{\mathsf{R}})^{[k]} \\ (\underline{\boldsymbol{A}}_\ell^{\mathsf{I}})^{[k]} \\ (\underline{\boldsymbol{A}}_\ell^{o})^{[k]} \end{bmatrix} \tag{5.15}$$

such that the approximation of the sub-mode on a given macro interval can be written as:

$\forall t \in I_{k,\ell}^M$,

$$\bar{s}_\ell(t) = \underline{n}_\ell^T \left(\underline{\underline{\boldsymbol{I}}} \otimes (\underline{\Psi}_\ell^{[k]})^T \right) \underline{\boldsymbol{A}}_\ell^{[k]} \tag{5.16}$$

with $\underline{\underline{\boldsymbol{I}}}$ a 3×3 identity matrix and $\otimes$ denotes the Kronecker product. The expression (5.13) can be rewritten as:

$$J = \sum_{k=1}^{N_\ell} \int_{I_{k,\ell}^M} r_\ell^2 + (\underline{\boldsymbol{A}}_\ell^{[k]})^T \left(\underline{n}_\ell \underline{n}_\ell^T \otimes \underline{\Psi}_\ell^{[k]} (\underline{\Psi}_\ell^{[k]})^T \right) \underline{\boldsymbol{A}}_\ell^{[k]} - 2(\underline{\boldsymbol{A}}_\ell^{[k]})^T \left(\underline{\underline{\boldsymbol{I}}} \otimes \underline{\Psi}_\ell^{[k]} \right) \underline{n}_\ell r_\ell \, dt \tag{5.17}$$

By minimizing J with respect to $\underline{\boldsymbol{A}}_\ell^{[k]}$, $\forall k \in [1, ..., N_\ell]$ we obtain:

$$\underline{\underline{\boldsymbol{W}}}_\ell \underline{\boldsymbol{A}}_\ell = \underline{\boldsymbol{f}}_\ell \tag{5.18}$$

where $\underline{\boldsymbol{A}}_\ell$ corresponds to the vector containing all the DOFs related to the macro FEM discretization over the whole temporal domain, this is:

$$\underline{\boldsymbol{A}}_\ell = \bigoplus_{k=1}^{N_\ell} \underline{\boldsymbol{A}}_\ell^{[k]} \tag{5.19}$$

and with the matrices and vector given by:

$$\underline{\underline{\boldsymbol{W}}}_\ell = \bigoplus_{k=1}^{N_\ell} \int_{I_{k,\ell}^M} \left(\underline{n}_\ell \underline{n}_\ell^T \otimes \underline{\Psi}_\ell^{[k]} (\underline{\Psi}_\ell^{[k]})^T \right) dt \quad , \quad \underline{\boldsymbol{f}}_\ell = \bigoplus_{k=1}^{N_\ell} \int_{I_{k,\ell}^M} \left(\underline{\underline{\boldsymbol{I}}} \otimes \underline{\Psi}_\ell^{[k]} \right) \underline{n}_\ell r_\ell(t) dt \tag{5.20}$$

where $\bigoplus_{k=1}^{N_\ell}$ stands for the assembly operator over the macro FEM discretization.

From the above developments, the macro functions can be obtained by solving the equation (5.18), however this equation is still in function of the characteristic period τ_ℓ. To obtain this period, we use (5.18) to rewrite the functional J as follows:

$$J = \int_I r_\ell^2 dt + \underline{\boldsymbol{A}}_\ell^T \underline{\underline{\boldsymbol{W}}}_\ell \underline{\boldsymbol{A}}_\ell - 2 \underline{\boldsymbol{A}}_\ell^T \underline{\boldsymbol{f}}_\ell \tag{5.21}$$

By re-injecting the value of $\underline{\boldsymbol{A}}_\ell = \underline{\underline{\boldsymbol{W}}}_\ell^{-1} \underline{\boldsymbol{f}}_\ell$ in equation (5.21) we obtain:

$$\{\tau_\ell\} = \underset{\tau_\ell > 0}{\arg\min}\, J(\tau_\ell) = \underset{\tau_\ell > 0}{\arg\min} \left[\textstyle\int_I r_\ell^2 dt - \underline{\boldsymbol{f}}_\ell^T \underline{\underline{\boldsymbol{W}}}_\ell^{-1} \underline{\boldsymbol{f}}_\ell \right] \tag{5.22}$$

Therefore the minimization of $J(\tau_\ell)$ is given by:

$$\{\tau_\ell\} = \underset{\tau_\ell > 0}{\arg\max} \left[\underline{\boldsymbol{f}}_\ell^T \underline{\underline{\boldsymbol{W}}}_\ell^{-1} \underline{\boldsymbol{f}}_\ell \right] \tag{5.23}$$

The maximization of equation (5.23) allows to determine the characteristic period of the multiscale approximation, thus completing the construction of the sub-mode "ℓ". However, the maximization of this expression is not straightforward and an iterative search for the characteristic period must be applied to maximize the expression, a tedious and costly process. For this reason in the next subsection an approximation is introduced that allows the direct calculation of the characteristic period, greatly simplifying the maximization problem.

5.2.1.1.1 Determination of the characteristic periods

As developed before, the characteristic periods are obtained such that the expression (5.23) is maximized. However, this maximization requires an iterative search process that makes the procedure tedious. In order to solve this issue, we approximate the matrix $\underline{\underline{\boldsymbol{W}}}_\ell$ by:

$$\underline{\underline{\boldsymbol{W}}}_\ell \approx \bigoplus_{k=1}^{N_\ell} \underline{\underline{\boldsymbol{I}}} \otimes \frac{1}{2} \int_{I_{k,\ell}^M} \underline{\Psi}_\ell^{[k]} (\underline{\Psi}_\ell^{[k]})^T dt \tag{5.24}$$

this approximation arise naturally due to the difference in time scales between the macro and micro temporal intervals. By using equation (5.24) into the expression of equation (5.23) we obtain:

$$[\underline{\boldsymbol{f}}_\ell^T \underline{\underline{\boldsymbol{W}}}_\ell^{-1} \underline{\boldsymbol{f}}_\ell] = \underline{\boldsymbol{A}}_\ell^T \underline{\underline{\boldsymbol{W}}}_\ell \underline{\boldsymbol{A}}_\ell \approx \frac{T}{N_\ell} \sum_{k=1}^{N_\ell} \sum_{l=1}^{4} \left[(\boldsymbol{A}_{l,\ell}^{\mathsf{R}})^{[k]} \right]^2 + \left[(\boldsymbol{A}_{l,\ell}^{\mathsf{I}})^{[k]} \right]^2 \tag{5.25}$$

where l denotes the DOFs on a macro element k (which are 4 due to the use of cubic Hermite polynomials shape functions). The right-hand term of the equation (5.25) involves the sum of the quadratic nodal values of the discretized macro functions, this means that the expression $\underline{\boldsymbol{f}}_\ell^T \underline{\underline{\boldsymbol{W}}}_\ell^{-1} \underline{\boldsymbol{f}}_\ell$ can be approximately identified with a Fourier transformation (FT) on that macro element, which means $\underline{\boldsymbol{f}}_\ell^T \underline{\boldsymbol{f}}_\ell \approx \underline{\boldsymbol{f}}_\ell^T \underline{\underline{\boldsymbol{W}}}_\ell^{-1} \underline{\boldsymbol{f}}_\ell$. In this sense, we have:

$\forall k \in [1, ..., N_\ell]$, $l \in [1, ..., 4]$,

$$\left[(\boldsymbol{A}_{l,\ell}^{\mathsf{R}})^{[k]} \right]^2 + \left[(\boldsymbol{A}_{l,\ell}^{\mathsf{I}})^{[k]} \right]^2 \propto |\mathsf{FT}(r_\ell(t) \Psi_{l,\ell}^{[k]})|^2$$

Therefore, the characteristic period τ_ℓ that maximizes equation (5.23) can be approximately taken as:

$$\{\tau_\ell\} = \underset{\tau_\ell > 0}{\arg\max} \sqrt{\sum_{k=1}^{N_\ell} \sum_{l=1}^{4} |\mathsf{FT}(r_\ell(t) \Psi_{l,\ell}^{[k]})|^2} \tag{5.26}$$

where in order to be independent of the macro discretization, we simply consider:

$$\{\tau_\ell\} = \underset{\tau_\ell > 0}{\arg\max} \ |\mathsf{FT}\left(r_\ell(t)\right)| \tag{5.27}$$

The expression (5.27) allows to easily determine the characteristic periods for each sub-mode by simply applying a Fourier transformation of the residual function $r_\ell(t)$. As stated before these developments are an approximative way for the determination of the characteristics periods, its only simplifies the iterative process necessary for the maximization of the equation (5.23).

Once the characteristic period has been determined, one needs to compute the macro functions to completely construct the sub-mode. This is done using classical temporal FE formulation and solving equation (5.18), however, certain considerations must be taken into account before obtaining them. These considerations are explained below.

5.2.1.1.2 Considerations of the macro discretization and resolution

As presented in section 5.1.2 (equation (5.2)), in order to ensure a multiscale behavior of each sub-mode the following micro-macro relation is chosen:

$$\frac{\Delta T_\ell}{\tau_\ell} = n_c \quad , \quad n_c \in \mathbb{N}$$

Added to the last condition, an integer number of cycles should exist into each macro time interval. This induce a macro discretization that is not in all the cases compatible with the total duration of the signal. To solve this issue, we allow the last macro element of the discretization of each sub-mode ℓ to be defined on an interval outside the time domain by just adding zero values in this zone, allowing the method to work without modifications as presented previously. Finally, the macro functions for all the sub-modes is not imposed to be null at $t = 0$ and $t = T$ even if the value of the studied signal is zero at these points, this consideration allows to obtain better temporal sub-modes that improve the final approximation and therefore obtain better characteristic times.

5.2.1.2 Error calculation

In order to measure the quality of the approximation, after the calculation of a new sub-mode, an error estimator is computed. In this way, if we consider that we have already computed the $k-1$ sub-modes of the decomposition ($s_{\ell-1}(t)$) and by remembering the definition of the residual function as $r_\ell(t) = s_{ref}(t) - s_{\ell-1}(t)$, we define the relative error at mode ℓ as follows:

$$e_\ell = 100\sqrt{\left(\frac{\int_I (r_\ell(t) - \bar{s}_\ell(t))^2 dt}{\int_I s_{ref}(t)^2 dt}\right)} \; [\%] \tag{5.28}$$

Each extra mode is calculated until the error converged to a desired value $e_\ell < e_c$ or until a given quantity of modes m_c is reached.

5.2.2 Synthesis of the signal approximation

A new method for the approximation of signals was presented. The main idea of the method consists in approximating a given signal by a sum of multiscale sub-modes such as to allow at the same time the determination of very important parameters called *characteristic periods*. These parameters are needed for the temporal multiscale approximation of the PGD functions at the LATIN method which will be introduced in section 5.3. Algorithm 2 summarizes the different steps of the proposed method.

Algorithm 2: Search of the best ℓ sub-modes with $\frac{\Delta T_\ell}{\tau_\ell} = n_c$

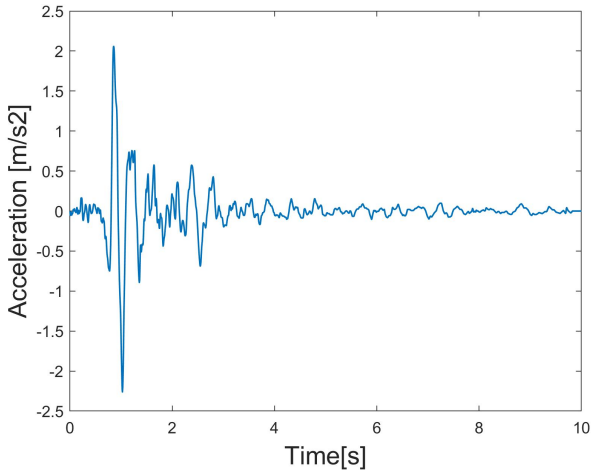
Input : $s_{ref}(t)$: Original signal, $m_c = 10$ (signal modes), e_c (stop error threshold)
Output : Characteristic periods $\{\tau_\ell\}_{\ell=1}^{m_s}$

$s_0(t) = 0;\ e_0 = 100;\ \ell = 0$
while $(\ell < m_c)$ **and** $(e_\ell > e_c)$ **do**
  $\ell = \ell + 1$
  Definition of the residual function: $\rightarrow r_\ell(t) = s(t) - s_{\ell-1}(t)$
  Characteristic period determination: $\tau_\ell = \underset{\tau_\ell > 0}{\arg\max}\ |\mathrm{FT}\,(r_\ell(t))|$
  Determination of the macro functions by solving: $\underline{\underline{\boldsymbol{W}}}_\ell \underline{\boldsymbol{A}}_\ell = \underline{\boldsymbol{f}}_\ell$
  Construction of the new signal sub-mode: $\bar{s}_\ell(t) = A_\ell^{\mathrm{R}}(t)\cos\left(\frac{2\pi t}{\tau_\ell}\right) + A_\ell^{\mathrm{I}}(t)\sin\left(\frac{2\pi t}{\tau_\ell}\right) + A_\ell^{o}(t)$
  Error calculation e_ℓ (equation (5.28)):
  Enrichment of the decomposition: $s_\ell(t) = s_{\ell-1}(t) + \bar{s}_\ell(t)$
end

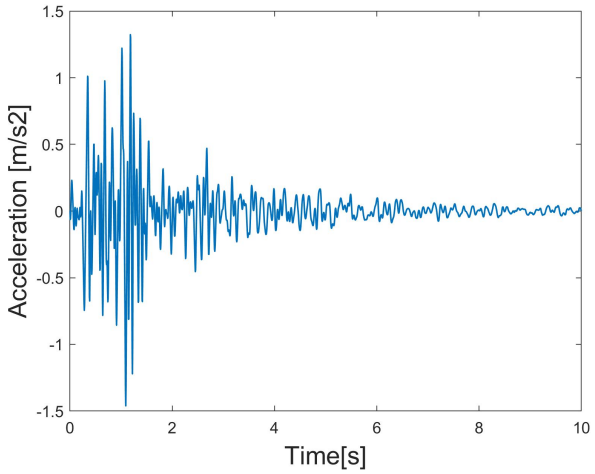
In order to show the performance of the presented signal modeling approach, the following section shows its application for the approximation of real seismic signals.

5.2.3 Numerical examples: approximation of seismic signals

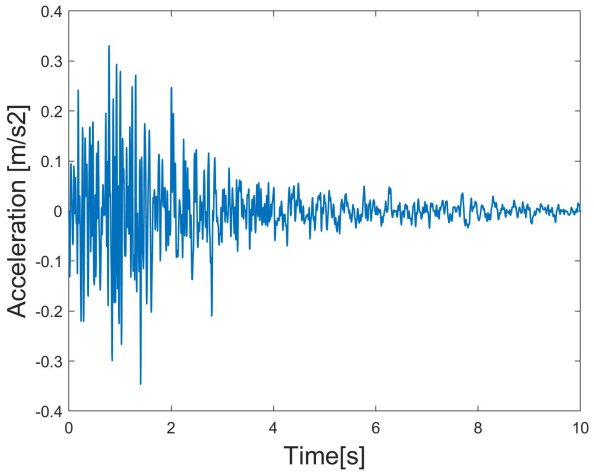
To test the signal approximation strategy introduced, we consider here the approximation of 3 different seismic signals in order to obtain the important *characteristic periods* associated with each of them. All signals are considered to have a duration of 10 seconds, these are presented in figures 5.9a, 5.9b and 5.9c. In addition, a micro-macro relation $\frac{\Delta T_\ell}{\tau_\ell} = 3$ is considered for all signals for they approximation, i.e., each sub-mode to be computed contains 3 cycles in a macro interval. This particular value for the micro-macro condition is arbitrary, however empirical numerical tests show that a number between 3 and 6 gives good results. The optimal value of the micro-macro ratio actually depends on the signal to be approximated, although its determination is complicated. For its correct choice several factors must be taken into account, for instance, a small ratio gives good results but the sub-modes lose their multiscale behavior and on the other hand, a large number could produce bad approximations. The optimal determination of the micro-macro ratio is still unknown and future research still should be done to numerically optimize its choice.



(a) Seismic signal 1.



(b) Seismic signal 2.



(c) Seismic signal 3.

Figure 5.9: Seismic signals considered for their approximation.

Once the multiscale approximation is applied to the signals, we obtain the results shown in figure 5.10.

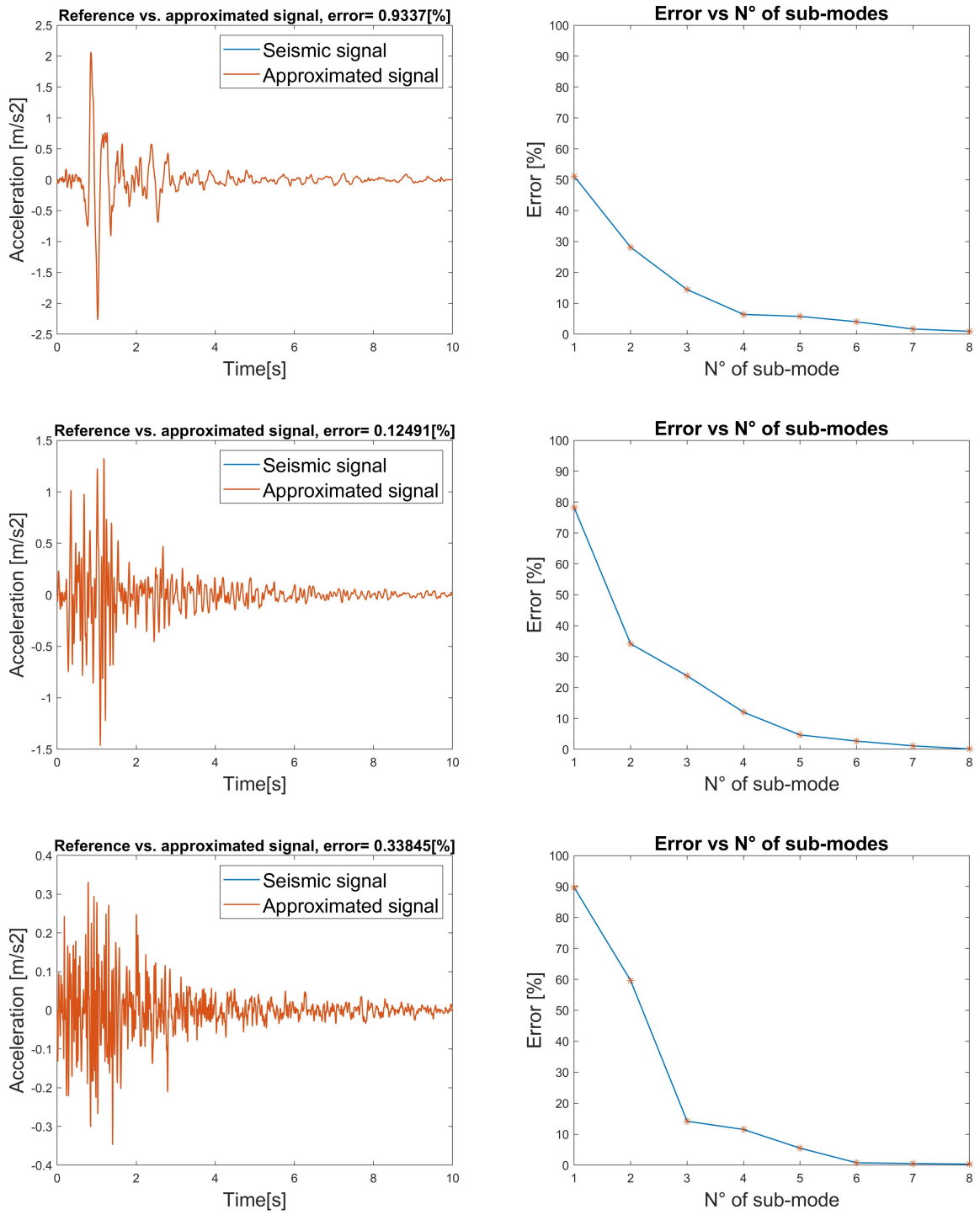
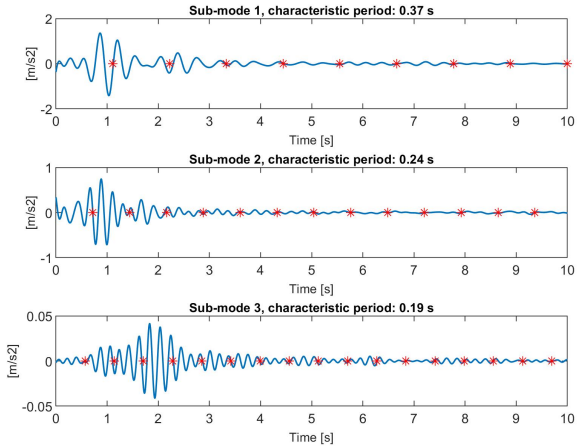


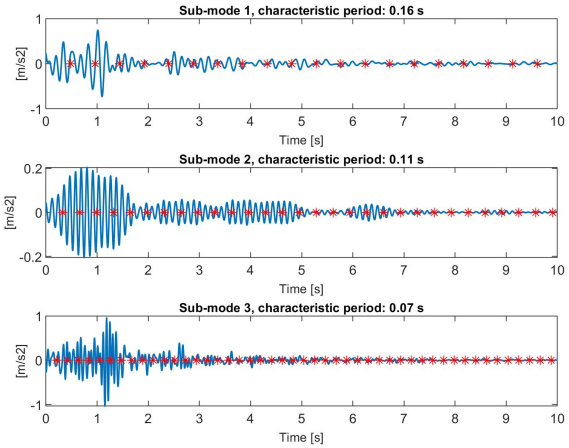
Figure 5.10: Comparison of reference signal to the approximated one at left, and the error vs sub-modes at right.

For all the signals studied, a maximum of 8 sub-modes are considered in order to present the convergence ratio for each of the signals. From the results in figure 5.10 we can observe that the multiscale strategy manages to approximate the considered signals very well, where very low errors are obtained even with a number of sub-modes less than 8, which verifies the high performance of the method.

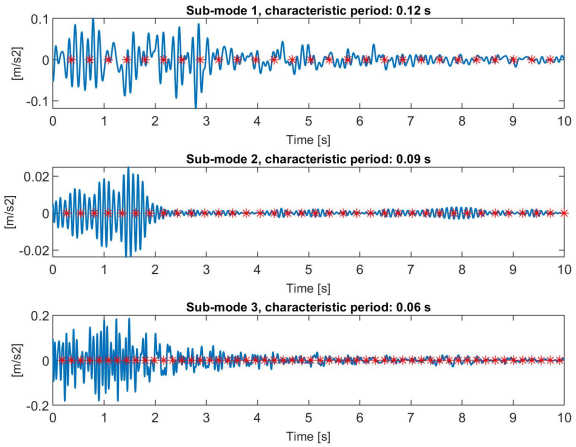
Figure 5.11 presents the first 3 multiscale sub-modes for the approximation of the seismic signals considered. From them we can clearly see the different macro discretization for each of the sub-modes, which is a function of its characteristic period, where the red dots indicate the FEM macro discretization.



(a) 3 first multiscale sub-modes of seismic signal 1.



(b) 3 first multiscale sub-modes of seismic signal 2.



(c) 3 first multiscale sub-modes of seismic signal 3.

Figure 5.11: First three multiscale sub-modes for the seismic signals considered.

It should be remembered that the main objective of the signal approximation method is to determine the characteristic periods of the external excitation signal. These periods are of great importance since they are used for the application of the multiscale approximation of the PGD time functions of the LATIN method. It should be remembered that the same amount of sub-modes determined from the external excitation are considered for the multiscale approximation of the temporal PGD functions, from this it can be concluded that the less sub-modes are required for the approximation of the external excitation, the faster the calculation of the temporal PGD functions will be.

All the details related to the new multiscale time approximation applied to the LATIN method are presented in the following sections, where the importance of the characteristic periods determined by the signal approximation discussed above will be clarified.

5.3 New temporal multiscale approach applied to the LATIN-PGD solver

As presented in chapter 4, the LATIN method is a global nonlinear solver which exploits the use of the model reduction technique PGD for enabling the decrease of the computational expenses. The PGD approximates the quantities of interest of the LATIN solver by a low-rank decomposition, in where space and temporal functions must be determined. This section focuses on the determination of the temporal PGD functions, where the new multiscale approximation introduced in equation 5.1 is applied. As presented in chapter 4, the calculation of the time functions is performed in two main steps which are the enrichment and the preliminary steps, where in both steps the new multiscale approach is applied. A brief explanation about the new multiscale approximation applied to these steps is given below, where important considerations for this approach are introduced.

- **Enrichment step**:

  This step consists in the calculation of a new space-time PGD couple function in order to enrich the reduced order model corresponding to the global solution. Its calculation is splitted into a space and a time problem, where in each one of them the spatial and temporal PGD function respectively are calculated. Once the spatial problem is solved, the time function is determined, which is approximated here in a multiscale manner as follows:

$$\lambda_{m+1}(t) \approx \sum_{\ell=1}^{m_s} \bar{\lambda}_\ell(t) = \sum_{\ell=1}^{m_s} A_\ell^{\mathsf{I}}(t) h_\ell^{\mathsf{I}}(t) + A_\ell^{\mathsf{R}}(t) h_\ell^{\mathsf{R}}(t) + A_\ell^{o}(t) \tag{5.29}$$

  Remark: It should be remembered that the number of sub-modes m_s and the characteristic periods $(\tau_\ell)_{\ell=1}^{m_s}$ used for the multiscale approximation of the temporal PGD functions are known since they have been determined when approximating the external excitation by the multiscale approximation explained in section 5.2.

  By defining the vector of micro functions $\underline{n}_\ell$ and the nodal values of the FEM approximation of the macro terms $\underline{\boldsymbol{A}}_\ell^{[k]}$ at the sub-mode "ℓ", as:

$$\underline{n}_\ell = \begin{bmatrix} h_\ell^{\mathsf{R}}(t) \\ h_\ell^{\mathsf{I}}(t) \\ 1 \end{bmatrix} \qquad ; \qquad \underline{\boldsymbol{A}}_\ell^{[k]} = \begin{bmatrix} (\underline{\boldsymbol{A}}_\ell^{\mathsf{R}})^{[k]} \\ (\underline{\boldsymbol{A}}_\ell^{\mathsf{I}})^{[k]} \\ (\underline{\boldsymbol{A}}_\ell^{o})^{[k]} \end{bmatrix}$$

  the discretized approximation defined on the macro interval $I_{k,\ell}^M$ can be written as:

  $\forall t \in I_{k,\ell}^M$,

$$\lambda_{m+1}(t) \approx \sum_{\ell=1}^{m_s} \underline{n}_\ell^T \left(\underline{\underline{\boldsymbol{I}}} \otimes (\underline{\Psi}_\ell^{[k]})^T \right) \underline{\boldsymbol{A}}_\ell^{[k]} \tag{5.30}$$

$$\dot{\lambda}_{m+1}(t) \approx \sum_{\ell=1}^{m_s} \dot{\underline{n}}_\ell^T \left(\underline{\underline{\boldsymbol{I}}} \otimes (\underline{\Psi}_\ell^{[k]})^T \right) \underline{\boldsymbol{A}}_\ell^{[k]} + \underline{n}_\ell^T \left(\underline{\underline{\boldsymbol{I}}} \otimes (\dot{\underline{\Psi}}_\ell^{[k]})^T \right) \underline{\boldsymbol{A}}_\ell^{[k]} \tag{5.31}$$

  where $\underline{\underline{\boldsymbol{I}}}$ represent a 3×3 identity matrix and $\otimes$ denotes the Kronecker product.

- **Preliminary step**:

  While the enrichment step consists in increasing the rank of the PGD decomposition, on the preliminary step the temporal PGD functions are actualized in order to improve the convergence to the nonlinear solution. This update is done by computing a group of corrective temporal functions such that:

$\forall i \in [1, ..., m+1]$,

$$\lambda_i \leftarrow \lambda_i + \Delta\lambda_i \tag{5.32}$$

these temporal corrections are also approximated in a multiscale way as follows:

$\forall i \in [1, ..., m+1]$, $\forall t \in I_{k,\ell}^M$,

$$\Delta\lambda_i(t) \approx \sum_{\ell=1}^{m_s} \Delta\bar{\lambda}_{i,\ell}(t) = \sum_{\ell=1}^{m_s} \underline{n}_{i,\ell}^T \left(\underline{\underline{\boldsymbol{I}}} \otimes (\underline{\Psi}_\ell^{[k]})^T \right) \underline{\boldsymbol{A}}_{i,\ell}^{[k]} \tag{5.33}$$

$$\Delta\dot{\lambda}_i(t) \approx \sum_{\ell=1}^{m_s} \Delta\dot{\bar{\lambda}}_{i,\ell}(t) = \sum_{\ell=1}^{m_s} \dot{\underline{n}}_{i,\ell}^T \left(\underline{\underline{\boldsymbol{I}}} \otimes (\underline{\Psi}_\ell^{[k]})^T \right) \underline{\boldsymbol{A}}_{i,\ell}^{[k]} + \underline{n}_{i,\ell}^T \left(\underline{\underline{\boldsymbol{I}}} \otimes (\dot{\underline{\Psi}}_\ell^{[k]})^T \right) \underline{\boldsymbol{A}}_{i,\ell}^{[k]} \tag{5.34}$$

with the micro functions and discretized vector of unknown variables associated to the macro functions at mode i given by:

$\forall i \in [1, ..., m+1]$,

$$\underline{n}_{i,\ell} = \begin{bmatrix} h_{i,\ell}^{\mathsf{R}}(t) \\ h_{i,\ell}^{\mathsf{I}}(t) \\ 1 \end{bmatrix} \quad ; \quad \underline{\boldsymbol{A}}_{i,\ell}^{[k]} = \begin{bmatrix} (\underline{\boldsymbol{A}}_{i,\ell}^{\mathsf{R}})^{[k]} \\ (\underline{\boldsymbol{A}}_{i,\ell}^{\mathsf{I}})^{[k]} \\ (\underline{\boldsymbol{A}}_{i,\ell}^{o})^{[k]} \end{bmatrix}$$

These compressed representations will be used along the following sections for the temporal functions associated to the isotropic damage and elasto-visco-plasticity which will be introduced later in this chapter.

• **Construction of the micro functions associated with the temporal PGD approximation**:

The micro functions used for the approximation of the PGD time functions are different compared to the signal approximation method presented in section 5.2, due to the nonlinear behavior of the material under consideration the micro functions are no longer simple sinusoids. Here they have a complex shape that must be determined at each enrichment or preliminary steps, however, they share the same characteristic period for all the sub-modes of the signal approximation method. The micro functions are constructed as the superposition of different functions, where each of them has a time domain equal to the characteristic time of the studied sub-mode. To this end, we introduce here a key notation that will be used throughout this section. Let us define the partition of a given function $f(t)$ as $[f(t)]_p$ such as:

$$f(t) = \bigcup_p [f(t)]_p \tag{5.35}$$

To visualize this idea, figure 5.12 illustrates the partitioning of a given function $f(t)$, defined on a time-segment of length τ_ℓ.

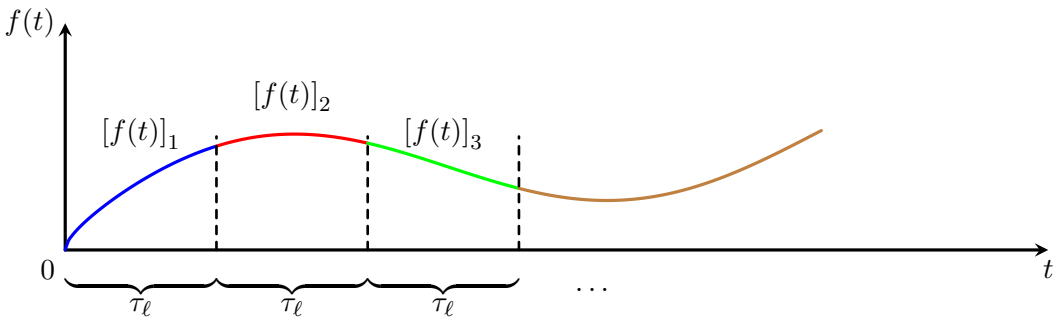


Figure 5.12: Partitioning of a function $f(t)$.

The function $f(t)$ used will depend on the treated problem and material nonlinearity as will be shown in the coming sections. From the above definitions the micro functions (symmetric and antisymmetric) are calculated as the superposition of functions, that is:

$$h_\ell^{\mathsf{R}}(t) = \sum_{p=1}^{N_{c,\ell}} a_p \left[f(t)\right]_{\mathsf{R},p} \qquad , \qquad h_\ell^{\mathsf{I}}(t) = \sum_{p=1}^{N_{c,\ell}} b_p \left[f(t)\right]_{\mathsf{I},p} \tag{5.36}$$

where $[f(t)]_{\mathsf{R},p}$ corresponds to the symmetric part of $[f(t)]_p$ and $[f(t)]_{\mathsf{I},p}$ to the antisymmetric part but with each beginning and end values equals to zero in order to ensure the continuity of each sub-mode (see section 5.1.2). From the above developments we can see that the main unknowns to be determined for the construction of the micro functions (apart from the good choice of $f(t)$) are the constants a_p and b_p $\forall p \in [1, ..., N_{c,\ell}]$.

The details about the determination of the macro and micro functions of the new multiscale approximation are given in sections 5.3.1 and 5.3.2 for the case of isotropic damage and elasto-visco-plasticity respectively. The presentation of the multiscale strategy for each material behavior is barely independent, since although the ideas are the same, the mathematical machinery behind is different.

Although in the present thesis the strategy is applied to two types of constitutive relations, it should be noted that its application to other types of behavior does not pose any difficulty.

5.3.1 Temporal multiscale approach applied to isotropic damage

On this section the new multiscale approximation in time is applied to the global stage when an isotropic damage constitutive relation is considered. The variables that must be determined at the global stage are the stress tensor $\underline{\underline{\sigma}}(\underline{x}, t)$, the total deformation tensor $\underline{\underline{\varepsilon}}(\underline{x}, t)$ and the damage variable $d(\underline{x}, t)$. Therefore, in the global stage at LATIN iteration $n+1$ the following solution set must be determined:

$$\mathcal{S}_{n+1} = \{\underline{u}_{n+1}, \underline{\underline{\varepsilon}}_{n+1}, \underline{\underline{\sigma}}_{n+1}, d_{n+1}\}$$

As presented in chapter 4, the global stage solution is computed such as it verifies the admissibility conditions and a descent search direction as illustrated in figure 5.13.

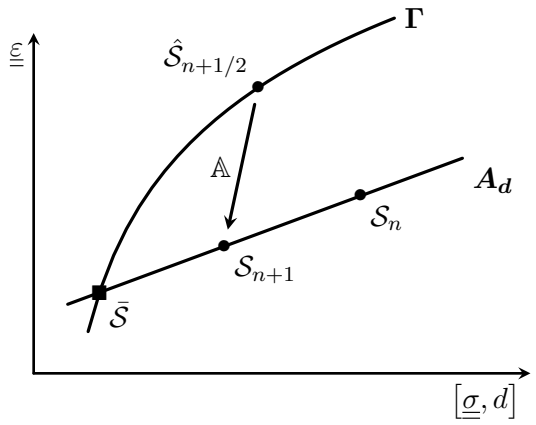


Figure 5.13: Calculation of the global solution of space $\boldsymbol{A_d}$.

This search direction is given by:

$$\mathbb{A}: \quad \begin{cases} d_{n+1} - \hat{d}_{n+1/2} = 0 \\ \left[\underline{\underline{\sigma}}_{n+1} - \hat{\underline{\underline{\sigma}}}_{n+1/2}\right] - \mathbb{H}_\varepsilon : \left[\underline{\underline{\varepsilon}}_{n+1} - \hat{\underline{\underline{\varepsilon}}}_{n+1/2}\right] = 0 \end{cases}$$

As exposed in chapter 4, instead of determining the full global stage quantities at iteration $n+1$ of the LATIN method, a corrective set solution is determined, which is given by:

$$\Delta\mathcal{S}_{n+1} = \{\Delta\underline{u}_{n+1}, \Delta\underline{\underline{\varepsilon}}_{n+1}, \Delta\underline{\underline{\sigma}}_{n+1}\}$$

so that it is verifies:

$$\mathcal{S}_{n+1} = \Delta\mathcal{S}_{n+1} + \mathcal{S}_n$$

From the above statements the following constitutive relation error (CRE) is defined:

$$J^c = \left|\left|\left|\Delta\underline{\underline{\sigma}}_{n+1} - \mathbb{H}_\varepsilon : \Delta\underline{\underline{\varepsilon}}_{n+1} + \underline{\underline{\Delta}}_{n+1}\right|\right|\right|^2_{\mathbb{H}_\varepsilon^{-1}} \quad , \quad |||\cdot|||^2_{\mathbb{H}_\varepsilon^{-1}} = \int_{\Omega\times I} (\cdot) : \mathbb{H}_\varepsilon^{-1} : (\cdot)\, d\Omega dt$$

with:

$$\underline{\underline{\Delta}}_{n+1} = (\underline{\underline{\sigma}}_n - \hat{\underline{\underline{\sigma}}}_{n+1/2})$$

The corrective terms are determined such as the CRE is minimized and the global stage solution is approached by employing the model reduction technique PGD, which allows the global variables to be written in a low-rank decomposition. In this sense, the global solution at LATIN iteration n ($\mathcal{S}_n$) is approximated as follows:

$$\begin{aligned} \underline{u}_n(\underline{x},t) &= \sum_{i=1}^{m} \bar{\underline{u}}_i(\underline{x})\lambda_i(t) + \underline{u}_0(\underline{x},t) \\ \underline{\underline{\varepsilon}}_n(\underline{x},t) &= \sum_{i=1}^{m} \bar{\underline{\underline{\varepsilon}}}_i(\underline{x})\lambda_i(t) + \underline{\underline{\varepsilon}}_0(\underline{x},t) \\ \underline{\underline{\sigma}}_n(\underline{x},t) &= \sum_{i=1}^{m} \bar{\underline{\underline{\sigma}}}_i(\underline{x})\lambda_i(t) + \underline{\underline{\sigma}}_0(\underline{x},t) \end{aligned}$$

where $\underline{u}_0(\underline{x},t)$, $\underline{\underline{\varepsilon}}_0(\underline{x},t)$ and $\underline{\underline{\sigma}}_0(\underline{x},t)$ correspond to the initial elastic solution and $\bar{\underline{u}}_i(\underline{x})$, $\bar{\underline{\underline{\sigma}}}_i(\underline{x})$, $\bar{\underline{\underline{\varepsilon}}}_i(\underline{x})$, $\lambda_i(t)$ correspond to the spatial and temporal PGD functions.

As mentioned previously, the PGD decomposition is constructed in the enrichment and preliminary steps. In the next sections the new multiscale approach is applied to the temporal problems associated with each of these steps.

5.3.1.1 Enrichment step: computation of the temporal PGD functions

For the approximation of the corrective solution set $\Delta\mathcal{S}_{n+1}$, the PGD is used, where a single PGD mode is calculated in order to approximate these corrections at the enrichment step, this is:

$$\begin{aligned} \Delta\underline{u}_{n+1} &= \underline{u}_{n+1} - \underline{u}_n = \bar{\underline{u}}_{m+1}(\underline{x})\lambda_{m+1}(t) \\ \Delta\underline{\underline{\varepsilon}}_{n+1} &= \underline{\underline{\varepsilon}}_{n+1} - \underline{\underline{\varepsilon}}_n = \bar{\underline{\underline{\varepsilon}}}_{m+1}(\underline{x})\lambda_{m+1}(t) \\ \Delta\underline{\underline{\sigma}}_{n+1} &= \underline{\underline{\sigma}}_{n+1} - \underline{\underline{\sigma}}_n = \bar{\underline{\underline{\sigma}}}_{m+1}(\underline{x})\lambda_{m+1}(t) \end{aligned} \tag{5.37}$$

This step is computed iteratively by solving a spatial and temporal problem as presented in section 4.3.2.1. Here we focus our attention on the temporal problem, where the multiscale approximation is applied. The

main problem consists in calculating a time function that minimizes the following minimization problem (which correspond to the constitutive relation error defined in chapter 4):

$$\{\lambda_{m+1}\} = \underset{\lambda_{m+1}\in\mathcal{U}^T}{\arg\min} \left|\left|\left| \underline{\underline{\bar{\sigma}}}_{m+1}\lambda_{m+1} - \mathbb{H}_\varepsilon : \underline{\underline{\bar{\varepsilon}}}_{m+1}\lambda_{m+1} + \underline{\underline{\Delta}}_{n+1} \right|\right|\right|^2_{\mathbb{H}_\varepsilon^{-1}} \tag{5.38}$$

In what follows the main calculation steps of the new multiscale approach are presented, which consists in the determination of the macro and micro time functions.

• **Macro functions determination**:

First of all the determination of the macro functions is explained, these macro terms depend on the choice of the micro functions, therefore this have to be seen as an intermediate process necessary to obtain both macro and micro functions of the multiscale decomposition. In this sense, lets consider the micro functions as a known variable and lets determine the best macro function such as it minimize the expression (5.38). Lets also remember that at this point the number of sub-modes m_s as well as the characteristic periods $(\tau_\ell)_{\ell=1}^{m_s}$ are known.

Due to the different sizes of the macro discretization, the temporal sub-modes are determined in an incremental way by a "greedy" process, this means, if we consider that the $\ell-1$ sub-modes have already been calculated, we can rewrite (5.38) as:

$$\left\{\bar{\lambda}_\ell\right\} = \underset{\bar{\lambda}_\ell\in\mathcal{U}^T}{\arg\min} \left|\left|\left| \underline{\underline{\bar{\sigma}}} \left(\sum_{s=1}^{\ell-1}\bar{\lambda}_s + \bar{\lambda}_\ell\right) - \mathbb{H}_\varepsilon : \underline{\underline{\bar{\varepsilon}}} \left(\sum_{s=1}^{\ell-1}\bar{\lambda}_s + \bar{\lambda}_\ell\right) + \underline{\underline{\Delta}}_{n+1} \right|\right|\right|^2_{\mathbb{H}_\varepsilon^{-1}} \tag{5.39}$$

where we consider $\underline{\underline{\bar{\sigma}}} = \underline{\underline{\bar{\sigma}}}_{m+1}$, $\underline{\underline{\bar{\varepsilon}}} = \underline{\underline{\bar{\varepsilon}}}_{m+1}$ and $\lambda = \lambda_{m+1}$ to alleviate the notations. By rearranging the terms of the minimization problem we obtain:

$$\left\{\bar{\lambda}_\ell\right\} = \underset{\bar{\lambda}_\ell\in\mathcal{U}^T}{\arg\min} \left|\left|\left| \underline{\underline{\bar{\sigma}}}\bar{\lambda}_\ell - \mathbb{H}_\varepsilon : \underline{\underline{\bar{\varepsilon}}}\bar{\lambda}_\ell + \underline{\underline{\bar{\Delta}}}^{\ell}_{n+1} \right|\right|\right|^2_{\mathbb{H}_\varepsilon^{-1}} \tag{5.40}$$

with the residual function given by:

$$\underline{\underline{\bar{\Delta}}}^{\ell}_{n+1} = \underline{\underline{\Delta}}_{n+1} + \left(\underline{\underline{\bar{\sigma}}} - \mathbb{H}_\varepsilon : \underline{\underline{\bar{\varepsilon}}}\right) \left(\sum_{s=1}^{\ell-1}\bar{\lambda}_s\right) \tag{5.41}$$

The minimization of equation (5.40) simply gives:

$\forall \delta\bar{\lambda}_\ell \in \mathcal{U}^T$,

$$\int_I \delta\bar{\lambda}_\ell \bar{\lambda}_\ell A^c \, dt = \int_I \delta\bar{\lambda}_\ell R^c_\ell(t) \, dt \tag{5.42}$$

where the constant scalar A^c is given by:

$$A^c = \int_\Omega \left(\underline{\underline{\bar{\sigma}}} - \mathbb{H}_\varepsilon : \underline{\underline{\bar{\varepsilon}}}\right) : \mathbb{H}_\varepsilon^{-1} : \left(\underline{\underline{\bar{\sigma}}} - \mathbb{H}_\varepsilon : \underline{\underline{\bar{\varepsilon}}}\right) d\Omega$$

and the residual temporal function by:

$$R^c_\ell(t) = D^c(t) - A^c \sum_{s=1}^{\ell-1}\bar{\lambda}_s(t) \qquad , \quad \text{with:} \quad D^c(t) = -\int_\Omega \left(\underline{\underline{\bar{\sigma}}} - \mathbb{H}_\varepsilon : \underline{\underline{\bar{\varepsilon}}}\right) : \mathbb{H}_\varepsilon^{-1} : \underline{\underline{\Delta}}_{n+1} \, d\Omega$$

By discretizing equation (5.42) using the multiscale approximation, we obtain:

$$\underline{\underline{\boldsymbol{Q}}}_\ell \, \underline{\boldsymbol{A}}_\ell = \underline{\boldsymbol{f}}_\ell$$

with the matrices and vector given by:

$$\underline{\underline{\boldsymbol{Q}}}_\ell = \bigoplus_{k=1}^{N_\ell} \int_{I_{k,\ell}^M} \left[\underline{n}_\ell \underline{n}_\ell^T \otimes \underline{\Psi}_\ell^{[k]} (\underline{\Psi}_\ell^{[k]})^T A^c \right] dt \qquad \text{and} \qquad \underline{\boldsymbol{f}}_\ell = \bigoplus_{k=1}^{N_\ell} \int_{I_{k,\ell}^M} \left[\left(\underline{\underline{\boldsymbol{I}}} \otimes \underline{\Psi}_\ell^{[k]} \right) \underline{n}_\ell R_\ell^c(t) \right] dt \tag{5.43}$$

So finally, the macro functions are simply determined as:

$$\underline{\boldsymbol{A}}_\ell = \underline{\underline{\boldsymbol{Q}}}_\ell^{-1} \underline{\boldsymbol{f}}_\ell \tag{5.44}$$

As mentioned above these macro functions still depend on the choice of the micro ones. The determination of these micro functions is presented below.

• **Micro functions determination**:

As exposed above, equation (5.44) still needs the determination of a good micro function that actually allows to decrease the constitutive relation error (5.40) , to do so, lets consider the CRE as follows:

$$J^c = \left| \left| \left| \underline{\underline{\bar{\sigma}}} \bar{\lambda}_\ell - \mathbb{H}_\varepsilon : \underline{\underline{\bar{\varepsilon}}} \bar{\lambda}_\ell + \underline{\underline{\bar{\Delta}}}_{n+1}^\ell \right| \right|_{\mathbb{H}_\varepsilon^{-1}}^2 \tag{5.45}$$

by expanding the functional J^c we have:

$$\begin{aligned} J^c = \int_{\Omega \times I} \underline{\underline{\bar{\Delta}}}_{n+1}^\ell : \mathbb{H}_\varepsilon^{-1} : \underline{\underline{\bar{\Delta}}}_{n+1}^\ell + \bar{\lambda}_\ell \bar{\lambda}_\ell \left(\underline{\underline{\bar{\sigma}}} - \mathbb{H}_\varepsilon : \underline{\underline{\bar{\varepsilon}}} \right) : \mathbb{H}_\varepsilon^{-1} : \left(\underline{\underline{\bar{\sigma}}} - \mathbb{H}_\varepsilon : \underline{\underline{\bar{\varepsilon}}} \right) \\ + 2 \bar{\lambda}_\ell \left(\underline{\underline{\bar{\sigma}}} - \mathbb{H}_\varepsilon : \underline{\underline{\bar{\varepsilon}}} \right) : \left(\mathbb{H}_\varepsilon^{-1} : \underline{\underline{\bar{\Delta}}}_{n+1}^\ell \right) d\Omega dt \end{aligned} \tag{5.46}$$

which can be simplified into:

$$J^c = \int_{\Omega \times I} \underline{\underline{\bar{\Delta}}}_{n+1}^\ell : \mathbb{H}_\varepsilon^{-1} : \underline{\underline{\bar{\Delta}}}_{n+1}^\ell d\Omega dt + \int_I \left[\bar{\lambda}_\ell^2 A^c - 2 \bar{\lambda}_\ell R_\ell^c(t) \right] dt \tag{5.47}$$

By introducing the macro and micro approximation into equation (5.47) we obtain the following expression:

$$J^c = \int_{\Omega \times I} \underline{\underline{\bar{\Delta}}}_{n+1}^\ell : \mathbb{H}_\varepsilon^{-1} : \underline{\underline{\bar{\Delta}}}_{n+1}^\ell d\Omega dt + \underline{\boldsymbol{A}}_\ell^T \underline{\underline{\boldsymbol{Q}}}_\ell \underline{\boldsymbol{A}}_\ell - 2 \underline{\boldsymbol{A}}_\ell^T \underline{\boldsymbol{f}}_\ell \tag{5.48}$$

By using the solution of the minimization of J^c with respect to the macro functions $\underline{\boldsymbol{A}}_\ell$ given in expression (5.44) and introducing it to the above equation we obtain:

$$J^c = \int_{\Omega \times I} \underline{\underline{\bar{\Delta}}}_{n+1}^\ell : \mathbb{H}_\varepsilon^{-1} : \underline{\underline{\bar{\Delta}}}_{n+1}^\ell d\Omega dt + \left(\underline{\underline{\boldsymbol{Q}}}_\ell^{-1} \underline{\boldsymbol{f}}_\ell \right)^T \underline{\underline{\boldsymbol{Q}}}_\ell \left(\underline{\underline{\boldsymbol{Q}}}_\ell^{-1} \underline{\boldsymbol{f}}_\ell \right) - 2 \left(\underline{\underline{\boldsymbol{Q}}}_\ell^{-1} \underline{\boldsymbol{f}}_\ell \right)^T \underline{\boldsymbol{f}}_\ell$$

which is reduced to:

$$J^c = \int_{\Omega \times I} \underline{\underline{\bar{\Delta}}}_{n+1}^\ell : \mathbb{H}_\varepsilon^{-1} : \underline{\underline{\bar{\Delta}}}_{n+1}^\ell d\Omega dt - \underline{\boldsymbol{f}}_\ell^T \underline{\underline{\boldsymbol{Q}}}_\ell^{-1} \underline{\boldsymbol{f}}_\ell$$

From the above expression of J^c we can conclude that:

$$\min J^c \quad \Longrightarrow \quad \max \underline{\boldsymbol{f}}_\ell^T \underline{\underline{\boldsymbol{Q}}}_\ell^{-1} \underline{\boldsymbol{f}}_\ell$$

this means that the micro functions should be determined such as:

$$\{\underline{n}_\ell\} = \max \ \underline{\boldsymbol{f}}_\ell^T \underline{\underline{\boldsymbol{Q}}}_\ell^{-1} \underline{\boldsymbol{f}}_\ell \tag{5.49}$$

The maximization of the equation (5.49) is complex since it requires knowledge of the matrix $\underline{\underline{\boldsymbol{Q}}}_\ell^{-1}$, which also depends on the micro functions. From the above we conclude that (5.49) must be solved iteratively, however, this increases the complexity of the strategy as well as the computational cost, both because of the iterative process and the inversion of the $\underline{\underline{\boldsymbol{Q}}}_\ell$ in each iteration. To solve this limitation, we approximate (5.49) as follows:

$$q_\ell = \underline{\boldsymbol{f}}_\ell^T \underline{\boldsymbol{f}}_\ell \approx \underline{\boldsymbol{f}}_\ell^T \underline{\underline{\boldsymbol{Q}}}_\ell^{-1} \underline{\boldsymbol{f}}_\ell \tag{5.50}$$

This approximation allows not only the direct determination of the micro functions, but also a considerable simplification for their determination. By neglecting the constant macro term (because it is not associated to a micro function) the above expression is written as:

$$q_\ell = \sum_{l=1}^{4} \sum_{k=1}^{N_\ell} \left[\int_{I_{k,\ell}^M} R_\ell^c(t) \Psi_{l,\ell}^{[k]} h_\ell^{\mathsf{R}}(t)\, dt \right]^2 + \left[\int_{I_{k,\ell}^M} R_\ell^c(t) \Psi_{l,\ell}^{[k]} h_\ell^{\mathsf{I}}(t)\, dt \right]^2 \tag{5.51}$$

where we recall the index l which corresponds to each component of the cubic Hermite shape functions. Although the equation (5.51) is the original functional to maximize, its maximization is numerically expensive due to the large number of operations to be performed. From the above we proceed to maximize an approximation of this functional, which is simply written as follows:

$$q_\ell \approx \sum_{l=1}^{4} \left[\int_I R_\ell^c(t) \Psi_{l,\ell} h_\ell^{\mathsf{R}}(t)\, dt \right]^2 + \left[\int_I R_\ell^c(t) \Psi_{l,\ell} h_\ell^{\mathsf{I}}(t)\, dt \right]^2 \tag{5.52}$$

where we denoted $\underline{\Psi}_\ell = \underline{\Psi}_\ell^{[k]}\ ,\ \forall k \in [1, ..., N_\ell]$ to simplify the notations (since the shape functions are the same due to the uniform macro discretization). This new expression eliminates the dependency on the macro discretization, greatly reducing the number of operations. This approximation is not new, indeed it was already used in the signal approximation technique in equation (5.27) to speed up the determination of the characteristic periods. This approximation will be used all along the present section to obtain the micro functions for the multiscale approach of the PGD functions.

The expression (5.52) still requires the definition of the micro functions. It is natural to realize that the temporal solution of the PGD will have a pattern similar to that of the residual time function $R_\ell^c(t)$, in this sense, it seems natural to construct the micro functions as follows:

$$h_\ell^{\mathsf{R}}(t) = \sum_{p=1}^{N_{c,\ell}} a_p^{(\ell)} \left[R_\ell^c(t) \right]_{\mathsf{R},p} \quad , \quad h_\ell^{\mathsf{I}}(t) = \sum_{p=1}^{N_{c,\ell}} b_p^{(\ell)} \left[R_\ell^c(t) \right]_{\mathsf{I},p} \tag{5.53}$$

By injecting the expression (5.53) into (5.52) we obtain:

$$q_\ell \approx \underbrace{\sum_{l=1}^{4} \left[\sum_{p_1=1}^{N_{c,\ell}} \int_{I_{p_1,\ell}} \left[R_\ell^c(t) \Psi_{l,\ell} \right]_{p_1} \sum_{p_2=1}^{N_{c,\ell}} a_{p_2}^{(\ell)} \left[R_\ell^c(t) \right]_{\mathsf{R},p_2} dt \right]^2}_{\underline{\boldsymbol{a}}_\ell^T \underline{\underline{\boldsymbol{N}}}_\ell^{\mathsf{R}} \underline{\boldsymbol{a}}_\ell} + \underbrace{\sum_{l=1}^{4} \left[\sum_{p_1=1}^{N_{c,\ell}} \int_{I_{p_1,\ell}} \left[R_\ell^c(t) \Psi_{l,\ell} \right]_{p_1} \sum_{p_2=1}^{N_{c,\ell}} b_{p_2}^{(\ell)} \left[R_\ell^c(t) \right]_{\mathsf{I},p_2} dt \right]^2}_{\underline{\boldsymbol{b}}_\ell^T \underline{\underline{\boldsymbol{N}}}_\ell^{\mathsf{I}} \underline{\boldsymbol{b}}_\ell} \tag{5.54}$$

with:

$$\underline{\boldsymbol{a}}_\ell = \begin{bmatrix} a_1^{(\ell)} \\ a_2^{(\ell)} \\ \vdots \\ a_{N_{c,\ell}} \end{bmatrix} \quad \text{and} \quad \underline{\boldsymbol{b}}_\ell = \begin{bmatrix} b_1^{(\ell)} \\ b_2^{(\ell)} \\ \vdots \\ b_{N_{c,\ell}}^{(\ell)} \end{bmatrix} \tag{5.55}$$

The maximization of equation (5.54) simply consists in determining the vectors $\underline{\boldsymbol{a}}_\ell$ and $\underline{\boldsymbol{b}}_\ell$ such as they maximize $\underline{\boldsymbol{a}}_\ell^T \underline{\underline{\boldsymbol{N}}}_\ell^{\mathsf{R}} \underline{\boldsymbol{a}}_\ell$ and $\underline{\boldsymbol{b}}_\ell^T \underline{\underline{\boldsymbol{N}}}_\ell^{\mathsf{I}} \underline{\boldsymbol{b}}_\ell$, therefore those vectors are easily selected as the eigenvector of matrices $\underline{\underline{\boldsymbol{N}}}_\ell^{\mathsf{R}}$ and $\underline{\underline{\boldsymbol{N}}}_\ell^{\mathsf{I}}$ associated to the their maximum eigenvalue.

Remark: Since we use the approximation (5.50), the micro functions are determined directly, this means that once the micro functions are determined, the macro terms are simply calculated using (5.44), which finalizes the multiscale approximation without the need for additional iterations.

5.3.1.2 Preliminary step: updating of the time functions

In this section, the preliminary resolution step, when applying the multiscale approximation is presented. As stated at the beginning of this section and in the last chapter, the main idea is to actualize the temporal PGD functions while maintaining the spatial one fixed. This process improves the convergence rate of the LATIN solver and avoids the computation of a large quantity of PGD modes. The main problem consists in finding the corrective temporal functions $\Delta\lambda_i \ \forall i \in [1, ..., m+1]$ such that they minimize the CRE given as follows (see chapter 4):

$$\{\Delta\lambda_i\}_{i=1}^{m+1} = \underset{\{\Delta\lambda_i\}_{i=1}^{m+1} \in \mathcal{U}^T}{\arg\min} \left| \left| \left| \sum_{i=1}^{m+1} \underline{\underline{\bar{\sigma}}}_i \Delta\lambda_i - \mathbb{H}_\varepsilon : \left(\sum_{i=1}^{m+1} \underline{\underline{\bar{\varepsilon}}}_i \Delta\lambda_i \right) + \underline{\underline{\Delta}}_{n+1} \right| \right|_{\mathbb{H}_\varepsilon^{-1}}^2$$

The idea herein consists in applying the new multiscale approximation for the determination of those corrective functions.

• **Macro functions determination**:

First of all lets consider the calculation of the macro functions. To do so lets assume that the first $\ell - 1$ sub-modes of the corrective functions have already been computed, this is:

$$\left\{\Delta\bar{\lambda}_{i,\ell}\right\}_{i=1}^{m+1} = \underset{\{\Delta\bar{\lambda}_{i,\ell}\}_{i=1}^{m+1} \in \mathcal{U}^T}{\arg\min} \left| \left| \left| \sum_{s=1}^{\ell-1} \sum_{i=1}^{m+1} \underline{\underline{\bar{\sigma}}}_i \left(\Delta\bar{\lambda}_{i,s}(t) + \Delta\bar{\lambda}_{i,\ell}(t)\right) - \mathbb{H}_\varepsilon : \sum_{s=1}^{\ell-1} \sum_{i=1}^{m+1} \underline{\underline{\bar{\varepsilon}}}_i \left(\Delta\bar{\lambda}_{i,s}(t) + \Delta\bar{\lambda}_{i,\ell}(t)\right) + \underline{\underline{\Delta}}_{n+1} \right| \right|_{\mathbb{H}_\varepsilon^{-1}}^2$$

By rearranging the known and unknown terms, the last expression can be reduced to the main problem that must be solved:

$$\left\{\Delta\bar{\lambda}_{i,\ell}\right\}_{i=1}^{m+1} = \underset{\{\Delta\bar{\lambda}_{i,\ell}\}_{i=1}^{m+1} \in \mathcal{U}^T}{\arg\min} \left| \left| \left| \sum_{i=1}^{m+1} \underline{\underline{\bar{\sigma}}}_i \Delta\bar{\lambda}_{i,\ell}(t) - \mathbb{H}_\varepsilon : \sum_{i=1}^{m+1} \underline{\underline{\bar{\varepsilon}}}_i \Delta\bar{\lambda}_{i,\ell}(t) + \underline{\underline{\bar{\Delta}}}_{n+1}^{\ell} \right| \right|_{\mathbb{H}_\varepsilon^{-1}}^2 \tag{5.56}$$

with the residual term given by:

$$\underline{\underline{\bar{\Delta}}}_{n+1}^{\ell} = \underline{\underline{\Delta}}_{n+1} + \sum_{s=1}^{\ell-1} \sum_{i=1}^{m+1} \left(\underline{\underline{\bar{\sigma}}}_i - \mathbb{H}_\varepsilon : \underline{\underline{\bar{\varepsilon}}}_i \right) \Delta\bar{\lambda}_{i,s}(t) \tag{5.57}$$

By minimizing the above expression we obtain:

$\forall \delta\Delta\bar{\lambda}_{i,\ell} \in \mathcal{U}^T, \ \forall i \in [1, ..., m+1]\,,$

$$\int_I \sum_{i=1}^{m+1} \sum_{j=1}^{m+1} \delta\Delta\bar{\lambda}_{i,\ell} \Delta\bar{\lambda}_{j,\ell} A_{ij}^c \ dt = \int_I \sum_{i=1}^{m+1} \delta\Delta\bar{\lambda}_{i,\ell} R_{i,\ell}^c(t) \ dt \tag{5.58}$$

where we recall below the definition of the following constants and temporal functions:

$\forall (i,j) \in [1,...,m+1]\,,$

$$A^c_{ij} = \int_\Omega \left(\bar{\underline{\underline{\sigma}}}_i - \mathbb{H}_\varepsilon : \bar{\underline{\underline{\varepsilon}}}_i\right) : \mathbb{H}_\varepsilon^{-1} : \left(\bar{\underline{\underline{\sigma}}}_j - \mathbb{H}_\varepsilon : \bar{\underline{\underline{\varepsilon}}}_j\right) d\Omega \quad \text{and} \quad D^c_i(t) = -\int_\Omega \left(\bar{\underline{\underline{\sigma}}}_i - \mathbb{H}_\varepsilon : \bar{\underline{\underline{\varepsilon}}}_i\right) : \mathbb{H}_\varepsilon^{-1} : \underline{\underline{\Delta}}_{n+1}\, d\Omega$$

and the residual functions:

$\forall i \in [1,...,m+1]\,,$

$$R^c_{i,\ell}(t) = D^c_i(t) - \sum_{s=1}^{\ell-1}\sum_{j=1}^{m+1} A^c_{ij}\bar{\lambda}_{j,s}(t) \tag{5.59}$$

By discretizing the temporal functions and introducing the new multiscale approximation (5.33), we obtain:

$$\underline{\underline{\boldsymbol{Q}}}_{up,\ell}\, \underline{\boldsymbol{A}}_{up,\ell} = \underline{\boldsymbol{f}}_{up,\ell} \tag{5.60}$$

where $\underline{\boldsymbol{A}}_{up,\ell}$ correspond to the nodal values in time of the macro functions associated to all the actualized modes at the sub-mode "ℓ".

$$\underline{\boldsymbol{A}}_{up,\ell} = \begin{bmatrix} \underline{\boldsymbol{A}}_{1,\ell} \\ \underline{\boldsymbol{A}}_{2,\ell} \\ \vdots \\ \\ \underline{\boldsymbol{A}}_{m+1,\ell} \end{bmatrix} \tag{5.61}$$

and the matrices and vectors given by:

$$\underline{\underline{\boldsymbol{Q}}}_{up,\ell} = \begin{bmatrix} \underline{\underline{\boldsymbol{Q}}}_{11,\ell} & \underline{\underline{\boldsymbol{Q}}}_{12,\ell} & \cdots & \underline{\underline{\boldsymbol{Q}}}_{1\ m+1,\ell} \\ \underline{\underline{\boldsymbol{Q}}}_{21,\ell} & \ddots & & \\ \vdots & & & \\ \underline{\underline{\boldsymbol{Q}}}_{m+1\ 1,\ell} & & & \underline{\underline{\boldsymbol{Q}}}_{m+1\ m+1,\ell} \end{bmatrix} \quad , \quad \underline{\boldsymbol{f}}_{up,\ell} = \begin{bmatrix} \underline{\boldsymbol{f}}_{1,\ell} \\ \underline{\boldsymbol{f}}_{2,\ell} \\ \vdots \\ \\ \underline{\boldsymbol{f}}_{m+1,\ell} \end{bmatrix}$$

with:

$\forall (i,j) \in [1,...,m+1]\,,$

$$\underline{\underline{\boldsymbol{Q}}}_{ij,\ell} = \bigoplus_{k=1}^{N_\ell} \int_{I^M_{k,\ell}} \underline{n}_{i,\ell}\underline{n}^T_{j,\ell} \otimes \underline{\Psi}^{[k]}_\ell (\underline{\Psi}^{[k]}_\ell)^T A^c_{ij}\, dt \qquad \text{and} \qquad \underline{\boldsymbol{f}}_{i,\ell} = \bigoplus_{k=1}^{N_\ell} \int_{I^M_{k,\ell}} \left(\underline{\underline{\boldsymbol{I}}} \otimes \underline{\Psi}^{[k]}_\ell\right) \underline{n}_{i,\ell} R^c_{i,\ell}(t)\, dt \tag{5.62}$$

• **Micro functions determination**:

Equation (5.60) gives the best macro functions for a given group of micro functions. The micro functions are determined following the same ideas as exposed in section 5.3.1.1 for the enrichment step. In this sense, we consider again the CRE (5.56):

$$J^c = \left|\left|\left| \sum_{i=1}^{m+1} \bar{\underline{\underline{\sigma}}}_i \Delta\bar{\lambda}_{i,\ell}(t) - \mathbb{H}_\varepsilon : \sum_{i=1}^{m+1} \bar{\underline{\underline{\varepsilon}}}_i \Delta\bar{\lambda}_{i,\ell}(t) + \bar{\underline{\underline{\Delta}}}^{\ell}_{n+1} \right|\right|\right|^2_{\mathbb{H}_\varepsilon^{-1}}$$

By developing the expression of J^c we have:

$$J^c = \int_{\Omega\times I} \bar{\underline{\underline{\Delta}}}^{\ell}_{n+1} : \mathbb{H}_\varepsilon^{-1} : \bar{\underline{\underline{\Delta}}}^{\ell}_{n+1}\, d\Omega dt + \sum_{i=1}^{m+1}\sum_{j=1}^{m+1} \Delta\bar{\lambda}_{i,\ell}(t)\Delta\bar{\lambda}_{j,\ell}(t) A^c_{ij} dt - 2\sum_{i=1}^{m+1} \Delta\bar{\lambda}_{i,\ell}(t) R^c_{i,\ell}(t) dt \tag{5.63}$$

By introducing the macro and micro approximation and using the result of equation (5.60) we obtain:

$$J^c = \int_{I\times\Omega} \underline{\underline{\bar{\Delta}}}^{\ell}_{n+1} : \mathbb{H}_{\varepsilon}^{-1} : \underline{\underline{\bar{\Delta}}}^{\ell}_{n+1}\, d\Omega dt - \underline{\boldsymbol{f}}^T_{up,\ell}\underline{\underline{\boldsymbol{Q}}}^{-1}_{up,\ell}\underline{\boldsymbol{f}}_{up,\ell}$$

From the above expression we can conclude that the minimization of J^c is achieved when the second term on the right is maximized, this is:

$$\text{min } J^c \quad \Longrightarrow \quad \text{max } \underline{\boldsymbol{f}}^T_{up,\ell}\underline{\underline{\boldsymbol{Q}}}^{-1}_{up,\ell}\underline{\boldsymbol{f}}_{up,\ell}$$

this means that the micro functions must be determined such as:

$\forall i \in [1, ..., m+1]$,

$$\left\{\underline{n}_{i,\ell}\right\} = \text{max } \underline{\boldsymbol{f}}^T_{up,\ell}\underline{\underline{\boldsymbol{Q}}}^{-1}_{up,\ell}\underline{\boldsymbol{f}}_{up,\ell}$$

As for the enrichment step case, the following approximation is considered in order to avoid an iterative calculation of the micro functions:

$$g_\ell = \underline{\boldsymbol{f}}^T_{up,\ell}\underline{\boldsymbol{f}}_{up,\ell} \approx \underline{\boldsymbol{f}}^T_{up,\ell}\underline{\underline{\boldsymbol{Q}}}^{-1}_{up,\ell}\underline{\boldsymbol{f}}_{up,\ell}$$

which can be simplified following the same ideas as for the enrichment step case as follows:

$$g_\ell \approx \sum_{i=1}^{m+1}\sum_{l=1}^{4}\left[\int_I R^c_{i,\ell}(t)\Psi_{l,\ell}h^{\mathsf{R}}_{i,\ell}(t)\, dt\right]^2 + \left[\int_I R^c_{i,\ell}(t)\Psi_{l,\ell}h^{\mathsf{I}}_{i,\ell}(t)\, dt\right]^2 \tag{5.64}$$

In the same way as in the enrichment step, we propose the micro functions to be constructed as follows:

$\forall i \in [1, ..., m+1]$,

$$h^{\mathsf{R}}_{i,\ell}(t) = \sum_{p=1}^{N_{c,\ell}} a^{(\ell)}_{i,p}\left[R^c_{i,\ell}(t)\right]_{\mathsf{R},p} \quad , \quad h^{\mathsf{I}}_{i,\ell}(t) = \sum_{p=1}^{N_{c,\ell}} b^{(\ell)}_{i,p}\left[R^c_{i,\ell}(t)\right]_{\mathsf{I},p} \tag{5.65}$$

By injecting the micro functions of expression (5.65) into (5.64) we finally obtain:

$$g_\ell \approx \sum_{i=1}^{m+1} \underline{\boldsymbol{a}}^T_{i,\ell}\underline{\underline{\boldsymbol{N}}}^{\mathsf{R}}_{i,\ell}\underline{\boldsymbol{a}}_{i,\ell} \;+\; \underline{\boldsymbol{b}}^T_{i,\ell}\underline{\underline{\boldsymbol{N}}}^{\mathsf{I}}_{i,\ell}\underline{\boldsymbol{b}}_{i,\ell} \tag{5.66}$$

with:

$\forall i \in [1, ..., m+1]$,

$$\underline{\boldsymbol{a}}^T_{i,\ell}\underline{\underline{\boldsymbol{N}}}^{\mathsf{R}}_{i,\ell}\underline{\boldsymbol{a}}_{i,\ell} = \sum_{l=1}^{4}\left[\sum_{p_1=1}^{N_{c,\ell}}\int_{I_{p_1,\ell}}\left[R^c_{i,\ell}(t)\Psi_{l,\ell}\right]_{p_1}\sum_{p_2=1}^{N_{c,\ell}} a^{(\ell)}_{i,p_2}\left[R^c_{i,\ell}(t)\right]_{\mathsf{R},p_2} dt\right]^2$$

$$\underline{\boldsymbol{b}}^T_{i,\ell}\underline{\underline{\boldsymbol{N}}}^{\mathsf{I}}_{i,\ell}\underline{\boldsymbol{b}}_{i,\ell} = \sum_{l=1}^{4}\left[\sum_{p_1=1}^{N_{c,\ell}}\int_{I_{p_1,\ell}}\left[R^c_{i,\ell}(t)\Psi_{l,\ell}\right]_{p_1}\sum_{p_2=1}^{N_{c,\ell}} b^{(\ell)}_{i,p_2}\left[R^c_{i,\ell}(t)\right]_{\mathsf{I},p_2} dt\right]^2$$

where the vector that contains the constants for each mode i that are defined as $\underline{\boldsymbol{a}}_{i,\ell}$ and $\underline{\boldsymbol{b}}_{i,\ell}$ are determined by calculating the eigenvector of matrices $\underline{\underline{\boldsymbol{N}}}^{\mathsf{R}}_{i,\ell}$ and $\underline{\underline{\boldsymbol{N}}}^{\mathsf{I}}_{i,\ell}$ associated to the their maximum eigenvalue.

5.3.2 Temporal multiscale approach applied to elasto-visco-plasticity

In the same way as presented for the case of isotropic damage, in this section the new multiscale approximation is applied to the global stage when considering elasto-visco-plasticity. For this material behavior the principal variables that must be determined correspond to the plastic deformation $\underline{\underline{\varepsilon}}^p(\underline{x},t)$, the stress $\underline{\underline{\sigma}}(\underline{x},t)$, total deformation $\underline{\underline{\varepsilon}}(\underline{x},t)$ and the internal variables $\underline{\underline{X}}(\underline{x},t)$, that are composed of the kinematic and isotropic hardening $\underline{\underline{\alpha}}$ and $\bar{r}$ respectively. Therefore, the quantities that must be determined at the global stage at LATIN iteration $n+1$ are given by the following solution set:

$$\mathcal{S}_{n+1} = \{\underline{u}_{n+1}, \underline{\underline{\varepsilon}}_{n+1}, \underline{\underline{\sigma}}_{n+1}, \underline{\underline{\varepsilon}}^p_{n+1}, \underline{\underline{\beta}}_{n+1}, \underline{\underline{\alpha}}_{n+1}, \bar{r}_{n+1}, \bar{R}_{n+1}\} \tag{5.67}$$

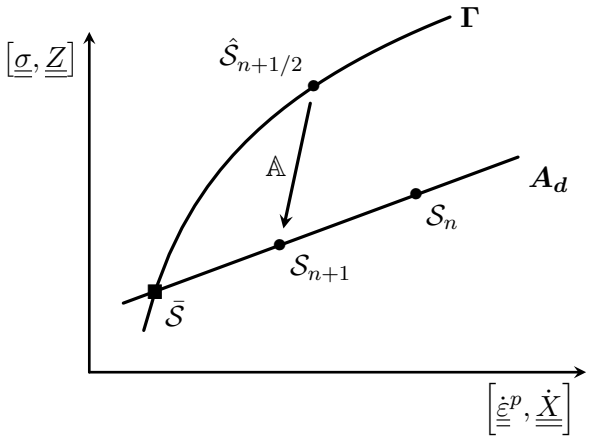


Figure 5.14: Calculation of the global solution of space $\boldsymbol{A_d}$.

This solution set must verify the admissibility conditions of the problem and at the same time the descent search direction as illustrated in figure 5.14, which is given below:

$$\mathbb{A}: \quad \left\{ \begin{bmatrix} \dot{\underline{\underline{\varepsilon}}}^p_{n+1} - \dot{\hat{\underline{\underline{\varepsilon}}}}^p_{n+1/2} \\ -(\dot{\underline{\underline{X}}}_{n+1} - \dot{\hat{\underline{\underline{X}}}}_{n+1/2}) \end{bmatrix} \right. - \mathbb{H}: \left. \begin{bmatrix} \underline{\underline{\sigma}}_{n+1} - \hat{\underline{\underline{\sigma}}}_{n+1/2} \\ \underline{\underline{Z}}_{n+1} - \hat{\underline{\underline{Z}}}_{n+1/2} \end{bmatrix} = 0 \right. \tag{5.68}$$

As presented in chapter 4, the unknown to be determined in the global stage corresponds to the corrective solution set $\Delta\mathcal{S}_{n+1}$, such that:

$$\mathcal{S}_{n+1} = \Delta\mathcal{S}_{n+1} + \mathcal{S}_n$$

with the corrective term solution given by:

$$\Delta\mathcal{S}_{n+1} = \{\Delta\underline{u}_{n+1}, \Delta\underline{\underline{\varepsilon}}_{n+1}, \Delta\underline{\underline{\sigma}}_{n+1}, \Delta\underline{\underline{\varepsilon}}^p_{n+1}, \Delta\underline{\underline{\beta}}_{n+1}, \Delta\underline{\underline{\alpha}}_{n+1}, \Delta\bar{r}_{n+1}, \Delta\bar{R}_{n+1}\} \tag{5.69}$$

This corrective solution set must be determined such that the distance between the global and local stages solution is minimized. To this end, by using the descent search direction, the following constitutive relation errors are defined:

$$J^p = \left|\left|\left| \Delta\dot{\underline{\underline{\varepsilon}}}^p_{n+1} - \mathbb{H}_\sigma : \Delta\underline{\underline{\sigma}}_{n+1} + \underline{\underline{\Delta}}_{n+1} \right|\right|\right|^2_{\mathbb{H}_\sigma^{-1}} \tag{5.70}$$

$$J^\alpha = \left|\left|\left| \Delta\dot{\underline{\underline{\alpha}}}_{n+1} + \mathbb{H}_\beta : (C\Delta\underline{\underline{\alpha}}_{n+1}) + \underline{\underline{\Delta}}^\alpha_{n+1} \right|\right|\right|^2_{\mathbb{H}_\beta^{-1}} \tag{5.71}$$

$$J^{\bar{r}} = \left|\left|\left| \Delta\dot{\bar{r}}_{n+1} + H_{\bar{R}} R_\infty \Delta\bar{r}_{n+1} + \Delta^{\bar{r}}_{n+1} \right|\right|\right|^2_{H_{\bar{R}}^{-1}} \tag{5.72}$$

with the residual terms and the used norms given by:

$$\begin{aligned} \underline{\underline{\Delta}}_{n+1} &= \dot{\underline{\underline{\varepsilon}}}^p_n - \hat{\dot{\underline{\underline{\varepsilon}}}}^p_{n+1/2} \\ \underline{\underline{\Delta}}^{\alpha}_{n+1} &= \dot{\underline{\underline{\alpha}}}_n - \hat{\dot{\underline{\underline{\alpha}}}}_{n+1/2} \\ \Delta^{\bar{r}}_{n+1} &= \dot{\bar{r}}_n - \hat{\dot{\bar{r}}}_{n+1/2} \end{aligned} \quad , \quad \begin{aligned} |||\cdot|||^2_{\mathbb{H}^{-1}_{\sigma}} &= \int_{\Omega\times I} (\cdot) : \mathbb{H}^{-1}_{\sigma} : (\cdot)\, d\Omega dt \\ |||\cdot|||^2_{\mathbb{H}^{-1}_{\beta}} &= \int_{\Omega\times I} (\cdot) : \mathbb{H}^{-1}_{\beta} : (\cdot)\, d\Omega dt \\ |||\cdot|||^2_{H^{-1}_{\bar{R}}} &= \int_{\Omega\times I} (\cdot) : H^{-1}_{\bar{R}} : (\cdot)\, d\Omega dt \end{aligned}$$

As for the isotropic damage case the global stage solution is calculated in an approximate and inexpensive way using the PGD, which allows to express the global quantities as a low-rank decomposition:

$$\begin{aligned} \underline{u}_n(\underline{x},t) &= \sum_{i=1}^{m} \underline{\bar{u}}_i(\underline{x})\lambda_i(t) + \underline{u}_0(\underline{x},t) \\ \underline{\underline{\varepsilon}}_n(\underline{x},t) &= \sum_{i=1}^{m} \underline{\underline{\bar{\varepsilon}}}_i(\underline{x})\lambda_i(t) + \underline{\underline{\varepsilon}}_0(\underline{x},t) \\ \underline{\underline{\sigma}}_n(\underline{x},t) &= \sum_{i=1}^{m} \underline{\underline{\bar{\sigma}}}_i(\underline{x})\lambda_i(t) + \underline{\underline{\sigma}}_0(\underline{x},t) \end{aligned} \quad , \quad \begin{aligned} \dot{\underline{\underline{\varepsilon}}}^p_n(\underline{x},t) &= \sum_{i=1}^{m} \dot{\underline{\underline{\bar{\varepsilon}}}}^p_i(\underline{x})\dot{\lambda}_i(t) \\ \dot{\underline{\underline{\alpha}}}_n(\underline{x},t) &= \sum_{i=1}^{m} \underline{\underline{\bar{\alpha}}}_i(\underline{x})\dot{\lambda}^{\alpha}_i(t) \\ \dot{\bar{r}}_n(\underline{x},t) &= \sum_{i=1}^{m} \bar{\bar{r}}_i(\underline{x})\dot{\lambda}^{\bar{r}}_i(t) \end{aligned}$$

As explained previously, this low-rank approximation is constructed at the *enrichment* and *preliminary* steps. The sections that follows present the temporal multiscale approximation applied to both steps. The procedure that will be exposed below follows the same ideas as exposed for the case of isotropic damage, however, the multiscale approximation applied to elasto-visco-plasticity is far more complicated since the constitutive relation is written in rate form. This makes necessary not only the correct construction of the micro and macro functions, but also their time derivatives. All these details are exposed below.

5.3.2.1 Enrichment step: computation of the temporal PGD functions

On the enrichment step, the rank of the PGD approximation is increased and only one PGD mode is determined. On this step a spatial and temporal problem must be solved as presented in previous chapters, however this section only focus on the resolution of the temporal problem by applying the new multiscale approximation. In this sense, the corrective terms of equation (5.69) are approximated by a rank-one decomposition, this is:

$$\begin{aligned} \Delta\underline{u}_{n+1} &= \underline{u}_{n+1} - \underline{u}_n = \underline{\bar{u}}_{m+1}(\underline{x})\lambda_{m+1}(t) \\ \Delta\underline{\underline{\varepsilon}}_{n+1} &= \underline{\underline{\varepsilon}}_{n+1} - \underline{\underline{\varepsilon}}_n = \underline{\underline{\bar{\varepsilon}}}_{m+1}(\underline{x})\lambda_{m+1}(t) \\ \Delta\underline{\underline{\sigma}}_{n+1} &= \underline{\underline{\sigma}}_{n+1} - \underline{\underline{\sigma}}_n = \underline{\underline{\bar{\sigma}}}_{m+1}(\underline{x})\lambda_{m+1}(t) \end{aligned} \quad , \quad \begin{aligned} \Delta\dot{\underline{\underline{\varepsilon}}}^p_{n+1} &= \dot{\underline{\underline{\varepsilon}}}^p_{n+1} - \dot{\underline{\underline{\varepsilon}}}^p_n = \dot{\underline{\underline{\bar{\varepsilon}}}}^p_{m+1}(\underline{x})\dot{\lambda}_{m+1}(t) \\ \Delta\dot{\underline{\underline{\alpha}}}_{n+1} &= \dot{\underline{\underline{\alpha}}}_{n+1} - \dot{\underline{\underline{\alpha}}}_n = \dot{\underline{\underline{\bar{\alpha}}}}_{m+1}(\underline{x})\dot{\lambda}^{\alpha}_{m+1}(t) \\ \Delta\dot{\bar{r}}_{n+1} &= \dot{\bar{r}}_{n+1} - \dot{\bar{r}}_n = \dot{\bar{\bar{r}}}_{m+1}(\underline{x})\dot{\lambda}^{\bar{r}}_{m+1}(t) \end{aligned} \tag{5.73}$$

The space and temporal PGD functions are determined such that they minimize the constitutive relation errors defined in equations (5.70), (5.71) and (5.72). The sections that follows present the calculation of the temporal PGD functions of equation (5.73) by applying the new multiscale approximation.

5.3.2.1.1 Temporal PGD function associated to the admissibility problem

Once the space functions $\underline{\bar{u}}_{m+1}$, $\underline{\underline{\bar{\sigma}}}_{m+1}$ and $\dot{\underline{\underline{\bar{\varepsilon}}}}^p_{m+1}$ of the corrective terms (5.73) are computed, the temporal function λ_{m+1} is calculated by minimizing the CRE (5.70), this is:

$$\{\lambda_{m+1}\} = \underset{\lambda_{m+1}\in\mathcal{U}^T}{\arg\min} \left|\left|\left| \dot{\underline{\underline{\bar{\varepsilon}}}}^p_{m+1}\dot{\lambda}_{m+1} - \mathbb{H}_{\sigma} : \underline{\underline{\bar{\sigma}}}_{m+1}\lambda_{m+1} + \underline{\underline{\Delta}}_{n+1} \right|\right|\right|^2_{\mathbb{H}^{-1}_{\sigma}} \tag{5.74}$$

In order to alleviate the notations in what follows we write $\lambda = \lambda_{m+1}$, $\underline{\underline{\bar{\varepsilon}}}^p = \underline{\underline{\bar{\varepsilon}}}^p_{m+1}$ and $\underline{\underline{\bar{\sigma}}} = \underline{\underline{\bar{\sigma}}}_{m+1}$. In the following the determination of the macro and micro functions of the multiscale approximation applied to the temporal PGD function are presented.

• **Macro functions determination**:

First, we will determine the macro functions. For this purpose, we consider that the $\ell - 1$ temporal sub-modes are already determined, therefore we can write:

$$\left\{\bar{\lambda}_\ell\right\} = \underset{\bar{\lambda}_\ell \in \mathcal{U}^T}{\arg\min} \left|\left|\left| \underline{\underline{\bar{\varepsilon}}}^p \left(\sum_{s=1}^{\ell-1} \dot{\bar{\lambda}}_s + \dot{\bar{\lambda}}_\ell \right) - \mathbb{H}_\sigma : \underline{\underline{\bar{\sigma}}} \left(\sum_{s=1}^{\ell-1} \bar{\lambda}_s + \bar{\lambda}_\ell \right) + \underline{\underline{\Delta}}_{n+1} \right|\right|\right|^2_{\mathbb{H}_\sigma^{-1}} \tag{5.75}$$

By rearranging the different terms, we obtain the problem to be solved:

$$\left\{\bar{\lambda}_\ell\right\} = \underset{\bar{\lambda}_\ell \in \mathcal{U}^T}{\arg\min} \left|\left|\left| \underline{\underline{\bar{\varepsilon}}}^p \dot{\bar{\lambda}}_\ell - \mathbb{H}_\sigma : \underline{\underline{\bar{\sigma}}} \bar{\lambda}_\ell + \underline{\underline{\bar{\Delta}}}^\ell_{n+1} \right|\right|\right|^2_{\mathbb{H}_\sigma^{-1}} \tag{5.76}$$

with:

$$\underline{\underline{\bar{\Delta}}}^\ell_{n+1} = \underline{\underline{\Delta}}_{n+1} + \sum_{s=1}^{\ell-1} \left(\underline{\underline{\bar{\varepsilon}}}^p \dot{\bar{\lambda}}_s - \mathbb{H}_\sigma : \underline{\underline{\bar{\sigma}}} \bar{\lambda}_s \right) \tag{5.77}$$

By developing the minimization problem we obtain:

$\forall \delta\bar{\lambda}_\ell \in \mathcal{U}^T$,

$$\int_I \delta\dot{\bar{\lambda}}_\ell \left(\dot{\bar{\lambda}}_\ell A^{11} + \bar{\lambda}_\ell A^{10} \right) + \delta\bar{\lambda}_\ell \left(\dot{\bar{\lambda}}_\ell A^{01} + \bar{\lambda}_\ell A^{00} \right) \, dt = \int_I \delta\dot{\bar{\lambda}}_\ell R^1_\ell(t) + \delta\bar{\lambda}_\ell R^0_\ell(t) \, dt \tag{5.78}$$

with the temporal residual functions given by:

$$R^1_\ell(t) = D^1(t) - \sum_{s=1}^{\ell-1} \left(\dot{\bar{\lambda}}_s A^{11} + \bar{\lambda}_s A^{10} \right) \tag{5.79}$$

$$R^0_\ell(t) = D^0(t) - \sum_{s=1}^{\ell-1} \left(\dot{\bar{\lambda}}_s A^{01} + \bar{\lambda}_s A^{00} \right) \tag{5.80}$$

where the constants and temporal functions are given by:

$$\begin{aligned} A^{11} &= \int_\Omega \underline{\underline{\bar{\varepsilon}}}^p : \mathbb{H}_\sigma^{-1} : \underline{\underline{\bar{\varepsilon}}}^p \, d\Omega \\ A^{10} &= -\int_\Omega \underline{\underline{\bar{\varepsilon}}}^p : \underline{\underline{\bar{\sigma}}} \, d\Omega \qquad\qquad D^1(t) = -\int_\Omega (\underline{\underline{\bar{\varepsilon}}}^p : \mathbb{H}_\sigma^{-1} : \underline{\underline{\Delta}}_{n+1}) \, d\Omega \\ A^{01} &= -\int_\Omega \underline{\underline{\bar{\sigma}}} : \underline{\underline{\bar{\varepsilon}}}^p \, d\Omega \qquad\qquad D^0(t) = \int_\Omega \underline{\underline{\bar{\sigma}}} : \underline{\underline{\Delta}}_{n+1} \, d\Omega \\ A^{00} &= \int_\Omega (\mathbb{H}_\sigma : \underline{\underline{\bar{\sigma}}}) : \underline{\underline{\bar{\sigma}}} \, d\Omega \end{aligned} \quad ,$$

By discretizing equation (5.78) using the multiscale approximation of the temporal PGD function (5.30) and its derivative (5.31), we obtain:

$$\underline{\underline{\boldsymbol{Q}}}_\ell \, \underline{\boldsymbol{A}}_\ell = \underline{\boldsymbol{f}}_\ell \tag{5.81}$$

with the discretized matrix and vector given by:

$$\begin{aligned}
&\underline{\underline{\boldsymbol{Q}}}_\ell = \\
&\bigoplus_{k=1}^{N_\ell} \int_{I^M_{k,\ell}} \left[\left(\dot{\underline{n}}_\ell \dot{\underline{n}}_\ell^T \otimes \underline{\Psi}_\ell^{[k]} (\underline{\Psi}_\ell^{[k]})^T \right) + \left(\dot{\underline{n}}_\ell \underline{n}_\ell^T \otimes \underline{\Psi}_\ell^{[k]} (\dot{\underline{\Psi}}_\ell^{[k]})^T \right) + \left(\underline{n}_\ell \dot{\underline{n}}_\ell^T \otimes \dot{\underline{\Psi}}_\ell^{[k]} (\underline{\Psi}_\ell^{[k]})^T \right) + \left(\underline{n}_\ell \underline{n}_\ell^T \otimes \dot{\underline{\Psi}}_\ell^{[k]} (\dot{\underline{\Psi}}_\ell^{[k]})^T \right) \right] A^{11} \\
&+ \left[\left(\dot{\underline{n}}_\ell \underline{n}_\ell^T \otimes \underline{\Psi}_\ell^{[k]} (\underline{\Psi}_\ell^{[k]})^T \right) + \left(\underline{n}_\ell \underline{n}_\ell^T \otimes \dot{\underline{\Psi}}_\ell^{[k]} (\underline{\Psi}_\ell^{[k]})^T \right) \right] A^{10} + \left[\left(\underline{n}_\ell \dot{\underline{n}}_\ell^T \otimes \underline{\Psi}_\ell^{[k]} (\underline{\Psi}_\ell^{[k]})^T \right) + \left(\underline{n}_\ell \underline{n}_\ell^T \otimes \underline{\Psi}_\ell^{[k]} (\dot{\underline{\Psi}}_\ell^{[k]})^T \right) \right] A^{01} \\
&+ \left[\underline{n}_\ell \underline{n}_\ell^T \otimes \underline{\Psi}_\ell^{[k]} (\underline{\Psi}_\ell^{[k]})^T \right] A^{00} \, dt
\end{aligned} \tag{5.82}$$

$$\underline{\boldsymbol{f}}_\ell = \bigoplus_{k=1}^{N_\ell} \int_{I^M_{k,\ell}} \left[\left(\underline{\underline{\boldsymbol{I}}} \otimes \underline{\Psi}_\ell^{[k]} \right) \dot{\underline{n}}_\ell + \left(\underline{\underline{\boldsymbol{I}}} \otimes \dot{\underline{\Psi}}_\ell^{[k]} \right) \underline{n}_\ell \right] R^1_\ell(t) + \left(\underline{\underline{\boldsymbol{I}}} \otimes \underline{\Psi}_\ell^{[k]} \right) \underline{n}_\ell R^0_\ell(t) \, dt \tag{5.83}$$

Of course, the above resolution still depends on the micro functions which have to be determined, this procedure is exposed below.

• **Micro functions determination**:

The discretized equation (5.81) gives the best macro functions for a given micro ones. The micro functions, on the other hand, are determined such that they minimize the CRE. In this sense, lets consider the error to minimize:

$$J^p = \left| \left| \left| \underline{\underline{\bar{\varepsilon}}}^p \dot{\bar{\lambda}}_\ell(t) - \mathbb{H}_\sigma : \underline{\underline{\bar{\sigma}}} \bar{\lambda}_\ell(t) + \underline{\underline{\bar{\Delta}}}^\ell_{n+1} \right| \right|^2_{\mathbb{H}^{-1}_\sigma}$$

which full expression is given by:

$$J^p = \int_{\Omega \times I} \left(\underline{\underline{\bar{\varepsilon}}}^p \dot{\bar{\lambda}}_\ell - \mathbb{H}_\sigma : \underline{\underline{\bar{\sigma}}} \bar{\lambda}_\ell + \underline{\underline{\bar{\Delta}}}^\ell_{n+1} \right) : \mathbb{H}^{-1}_\sigma : \left(\underline{\underline{\bar{\varepsilon}}}^p \dot{\bar{\lambda}}_\ell - \mathbb{H}_\sigma : \underline{\underline{\bar{\sigma}}} \bar{\lambda}_\ell + \underline{\underline{\bar{\Delta}}}^\ell_{n+1} \right) d\Omega dt \tag{5.84}$$

By developing and simplifying the terms we obtain:

$$\begin{aligned}
J^p = &\int_{\Omega \times I} \underline{\underline{\bar{\Delta}}}^\ell_{n+1} : \mathbb{H}^{-1}_\sigma : \underline{\underline{\bar{\Delta}}}^\ell_{n+1} \, d\Omega dt + \int_I \dot{\bar{\lambda}}_\ell \dot{\bar{\lambda}}_\ell A^{11}(t) + \dot{\bar{\lambda}}_\ell \bar{\lambda}_\ell A^{10} + \bar{\lambda}_\ell \dot{\bar{\lambda}}_\ell A^{10} + \bar{\lambda}_\ell \bar{\lambda}_\ell A^{00} \\
&- 2 \left(\dot{\bar{\lambda}}_\ell R^1_\ell(t) + \bar{\lambda}_\ell R^0_\ell(t) \right) \, dt
\end{aligned} \tag{5.85}$$

By introducing the discretized multiscale approximation of the temporal PGD function along with the solution of the macro functions that minimize the CRE into the above expression we obtain:

$$J^p = \int_{\Omega \times I} \underline{\underline{\bar{\Delta}}}^\ell_{n+1} : \mathbb{H}^{-1}_\sigma : \underline{\underline{\bar{\Delta}}}^\ell_{n+1} \, d\Omega dt - \underline{\boldsymbol{f}}^T_\ell \underline{\underline{\boldsymbol{Q}}}^{-1}_\ell \underline{\boldsymbol{f}}_\ell \tag{5.86}$$

The above expression is equivalent to the one obtained in the isotropic damage case, therefore the micro functions must be calculated such that:

$$\{\underline{n}_\ell\} = \max \; \underline{\boldsymbol{f}}^T_\ell \underline{\underline{\boldsymbol{Q}}}^{-1}_\ell \underline{\boldsymbol{f}}_\ell$$

As explained for the case of isotropic damage, the following approach is taken to calculate the micro functions directly (to avoid an iterative resolution strategy by avoiding the term $\underline{\underline{\boldsymbol{Q}}}^{-1}_\ell$):

$$q_\ell = \underline{\boldsymbol{f}}^T_\ell \underline{\boldsymbol{f}}_\ell \approx \underline{\boldsymbol{f}}^T_\ell \underline{\underline{\boldsymbol{Q}}}^{-1}_\ell \underline{\boldsymbol{f}}_\ell$$

Whose approximation is given by:

$$q_\ell \approx \sum_{l=1}^{4} \left[\int_I (\dot{h}_\ell^{\mathsf{R}} \Psi_{l,\ell} + h_\ell^{\mathsf{R}} \dot{\Psi}_{l,\ell}) R_\ell^1(t) + h_\ell^{\mathsf{R}} \Psi_{l,\ell} R_\ell^0(t)\, dt \right]^2 + \left[\int_I (\dot{h}_\ell^{\mathsf{I}} \Psi_{l,\ell} + h_\ell^{\mathsf{I}} \dot{\Psi}_{l,\ell}) R_\ell^1(t) + h_\ell^{\mathsf{I}} \Psi_{l,\ell} R_\ell^0(t)\, dt \right]^2 \tag{5.87}$$

It should be noted that the expression (5.87) implies the use of the micro functions and their derivatives in time, which increases the complexity for their determination. If we follow the same argument stated for the isotropic damage case, here we find two possible candidate functions for the construction of the micro terms, $R_\ell^1(t)$ and $R_\ell^0(t)$, both residual functions given in (5.79) and (5.80) respectively. Both can be chosen, however, in this thesis work we consider the function $R_\ell^1(t)$, this implies that:

$$\begin{aligned} h_\ell^{\mathsf{R}}(t) &= \sum_{p=1}^{N_{c,\ell}} a_p^{(\ell)} \int \left[R_\ell^1(t)\right]_{\mathsf{I},p} dt & h_\ell^{\mathsf{I}}(t) &= \sum_{p=1}^{N_{c,\ell}} b_p^{(\ell)} \int \left[R_\ell^1(t)\right]_{\mathsf{R},p} dt \\ \dot{h}_\ell^{\mathsf{R}}(t) &= \sum_{p=1}^{N_{c,\ell}} a_p^{(\ell)} \left[R_\ell^1(t)\right]_{\mathsf{I},p} & \dot{h}_\ell^{\mathsf{I}}(t) &= \sum_{p=1}^{N_{c,\ell}} b_p^{(\ell)} \left[R_\ell^1(t)\right]_{\mathsf{R},p} \end{aligned} \tag{5.88}$$

By injecting the expression of the micro functions (5.88) into (5.87) we finally obtain:

$$\begin{aligned} q_\ell \approx & \underbrace{\sum_{l=1}^{4} \left[\sum_{p_1=1}^{N_{c,\ell}} \int_{I_{p_1}} \left[\Psi_{l,\ell} R_\ell^1(t)\right]_{p_1} \sum_{p_2=1}^{N_{c,\ell}} a_{p_2}^{(\ell)} \left[R_\ell^1(t)\right]_{\mathsf{I},p_2} + \left[\dot{\Psi}_{l,\ell} R_\ell^1(t) + \Psi_{l,\ell} R_\ell^0(t)\right]_{p_1} \left(\sum_{p_2=1}^{N_{c,\ell}} a_{p_2}^{(\ell)} \int \left[R_\ell^1(t)\right]_{\mathsf{I},p_2} dt \right) dt \right]^2}_{\underline{\boldsymbol{a}}_\ell^T \underline{\underline{\boldsymbol{N}}}_\ell^{\mathsf{R}} \underline{\boldsymbol{a}}_\ell} \\ & + \underbrace{\sum_{l=1}^{4} \left[\sum_{p_1=1}^{N_{c,\ell}} \int_{I_{p_1}} \left[\Psi_{l,\ell} R_\ell^1(t)\right]_{p_1} \sum_{p_2=1}^{N_{c,\ell}} b_{p_2}^{(\ell)} \left[R_\ell^1(t)\right]_{\mathsf{R},p_2} + \left[\dot{\Psi}_{l,\ell} R_\ell^1(t) + \Psi_{l,\ell} R_\ell^0(t)\right]_{p_1} \left(\sum_{p_2=1}^{N_{c,\ell}} b_{p_2}^{(\ell)} \int \left[R_\ell^1(t)\right]_{\mathsf{R},p_2} dt \right) dt \right]^2}_{\underline{\boldsymbol{b}}_\ell^T \underline{\underline{\boldsymbol{N}}}_\ell^{\mathsf{I}} \underline{\boldsymbol{b}}_\ell} \end{aligned} \tag{5.89}$$

where the vectors $\underline{\boldsymbol{a}}_\ell$ and $\underline{\boldsymbol{b}}_\ell$ are determined by calculating the eigenvector of matrices $\underline{\underline{\boldsymbol{N}}}_\ell^{\mathsf{R}}$ and $\underline{\underline{\boldsymbol{N}}}_\ell^{\mathsf{I}}$ associated to the their maximum eigenvalue. Once the unknown vectors determined, the micro functions are completely defined, and therefore the macro functions can be computed using equation (5.81).

5.3.2.1.2 Internal variables

As shown in chapter 4, the temporal PGD functions related to the internal variables are also determined such that they minimize their respective CRE:

$$\{\lambda_{m+1}^{\alpha}\} = \underset{\lambda_{m+1}^{\alpha} \in \mathcal{U}^T}{\arg\min} \left| \left| \left| \underline{\underline{\bar{\alpha}}}_{m+1} \dot{\lambda}_{m+1}^{\alpha} + \mathbb{H}_\beta : C \underline{\underline{\bar{\alpha}}}_{m+1} \lambda_{m+1}^{\alpha} + \underline{\underline{\Delta}}_{n+1}^{\alpha} \right| \right|_{\mathbb{H}_\beta^{-1}}^2 \tag{5.90}$$

$$\{\lambda_{m+1}^{\bar{r}}\} = \underset{\lambda_{m+1}^{\bar{r}} \in \mathcal{U}^T}{\arg\min} \left| \left| \left| \bar{\bar{r}}_{m+1} \dot{\lambda}_{m+1}^{\bar{r}} + H_{\bar{R}} R_\infty \bar{\bar{r}}_{m+1} \lambda_{m+1}^{\bar{r}} + \Delta_{n+1}^{\bar{r}} \right| \right|_{H_{\bar{R}}^{-1}}^2 \tag{5.91}$$

The resolution of the above problems is completely equivalent to the resolution of the temporal function associated to the admissibility problem (λ_{m+1}), therefore the presentation of the temporal multiscale approximation applied to the internal variables quantities is omitted.

5.3.2.2 Preliminary step: updating of the time functions

The development of the macro and micro problems in the preliminary stage applied to elasto-visco-plasticity follows the same rules exposed for the isotropic damage case; in this sense, in order to ease the presentation of the multiscale strategy, the details of the preliminary stage applied to elasto-visco-plasticity are presented in the appendix D.

5.4 Numerical examples

This section presents numerical examples illustrating the new multiscale time approach, which is compared with the LATIN-PGD version using the Time Discontinuous Galerkin method presented in chapter 4. These examples consider isotropic damage and elasto-visco-plastic material behaviors, where in both cases the same 3D bending beam used in chapter 4 is considered, this in order to simplify the test and put a special focus on the new strategy. The numerical tests therefore consist in a beam where a displacement is imposed in both-end sides as shown in figure 5.15.

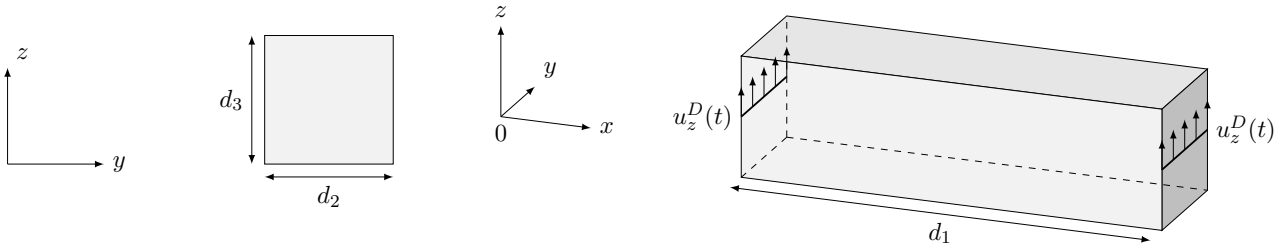


Figure 5.15: Test case considered, along with its dimensions.

We recall (as given in chapter 4) that the beam dimensions for the isotropic damage case are $d_1 = 8$ [m], $d_2 = d_3 = 0.3$ [m], while for elasto-visco-plasticity they are $d_1 = 6$ [m], $d_2 = d_3 = 0.1$ [m].

The purpose of the following results is entirely academic, where signals of relatively short duration in time are considered, the latter in order to better illustrate the strategy. More computationally intensive cases are left as a perspective, as many optimizations still need to be developed. The following results consider the following micro-macro relation:

$$\frac{\Delta T_\ell}{\tau_\ell} = 3 \ , \ \forall \ell \text{ (sub-mode)}$$

The LATIN thresholds considered to converge are chosen differently for the case of isotropic damage and elasto-visco-plasticity. This because isotropic damage requires a lower indicator than the elasto-visco-plasticity to produce accurate results compared to classical nonlinear incremental solvers as shown in chapter 4. In this sense, the following table resumes the LATIN threshold considered for each case.

Constitutive relation	LATIN error threshold
Isotropic damage	0.5 [%]
Elasto-visco-plasticity	1 [%]

Table 5.1: LATIN error threshold considered for each material behavior.

On the other hand, in the cases where a complex excitation is considered, the error threshold for the signal approximation (to determine the characteristic periods) is considered equal to 2 [%]. This choice is based

on empirical results, which show that a good signal approximation can be obtained while maintaining a low number of sub-modes.

5.4.1 Isotropic damage example

For the exemplification of the method, herein we will consider as first example the same monoperiodic excitation as presented in section 4.6.1 but with a smoother increase of the amplitude to allow a better multiscale approximation. This imposed displacement is shown in figure 5.16.

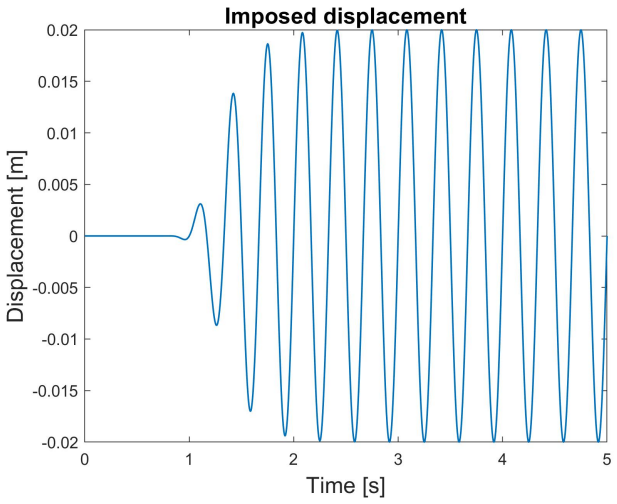


Figure 5.16: Imposed displacement.

The multiscale signal approximation for this case requires only one temporal sub-mode, with a characteristic period equal to the excitation period, which is given by:

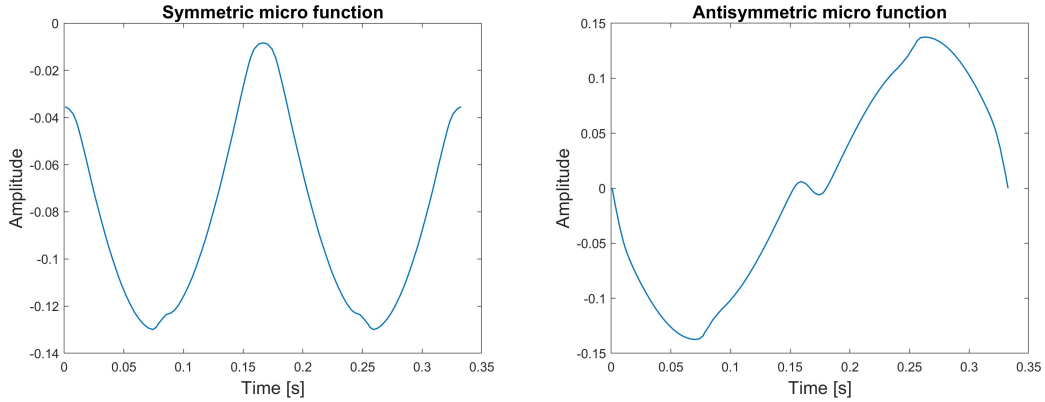
$$\tau = 0.33 \ [s]$$

Under these conditions, the DOFs of the temporal domain associated to each of the classic and multiscale version of the LATIN-PGD are presented in table 5.2. These DOFs are separated in the FEM for the classical discretization of the temporal domain and the DOFs related to the resolution of the macro and micro functions of the multiscale strategy.

Solver	FEM DOFs	DOFs of micro problem (Sym + Asym)
Classic LATIN-PGD	1800	0
Multiscale LATIN-PGD	36	24

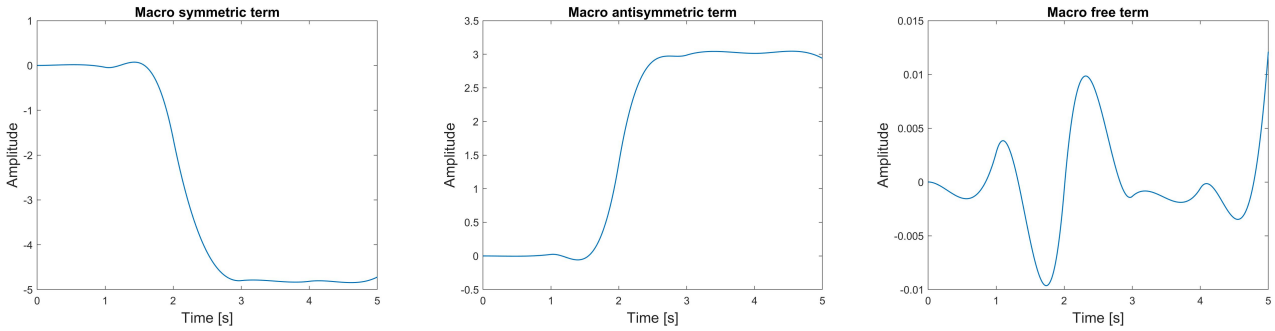
Table 5.2: Temporal DOFs.

When applying the multiscale approximation for the resolution of the first temporal PGD mode, a group of micro and macro functions are obtained. In this simple case figure 5.17 shows the symmetric and antisymmetric micro functions obtained for the first temporal PGD mode. These plots only show the function over a temporal interval equal to the characteristic time for a better visualization of them, which in this case corresponds to 0.33 [s], however the micro functions are defined and repeated all along the whole time interval. The macro functions in turn are shown in figure 5.18.



(a) Micro symmetric function. (b) Micro antisymmetric function.

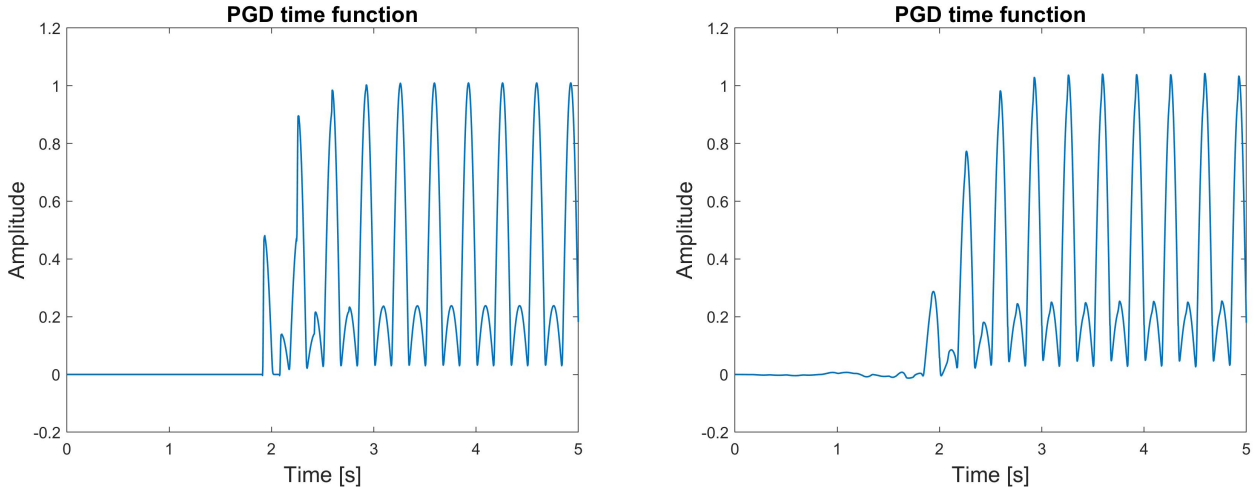
Figure 5.17: Micro functions.



(a) Macro symmetric term. (b) Macro antisymmetric term. (c) Macro free term.

Figure 5.18: Macro functions.

A comparison between the classic and the multiscale time functions are exposed in figure 5.19.



(a) Classic PGD time function. (b) Multiscale PGD time function.

Figure 5.19: Comparison between the classic and the multiscale PGD approximation.

Even if the temporal multiscale approximation introduced some errors due to the multiscale interpolation, the iterative resolution of the LATIN-PGD makes this approach converge towards the final solution without difficulty. This can be seen in figure 5.20 where a comparison of both resolution strategies is shown for the evolution of the damage variable over the most solicited integration point in space.

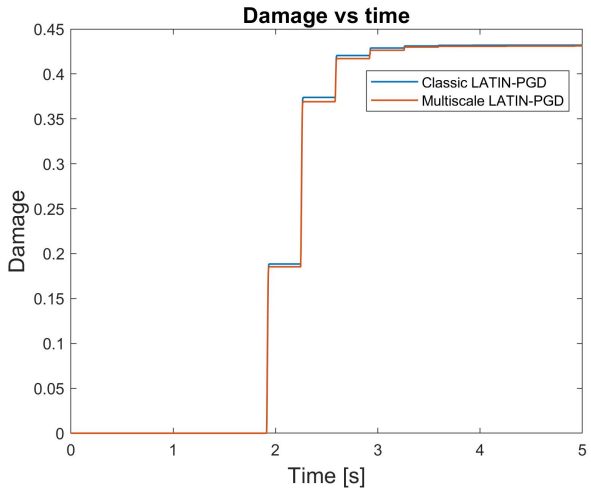
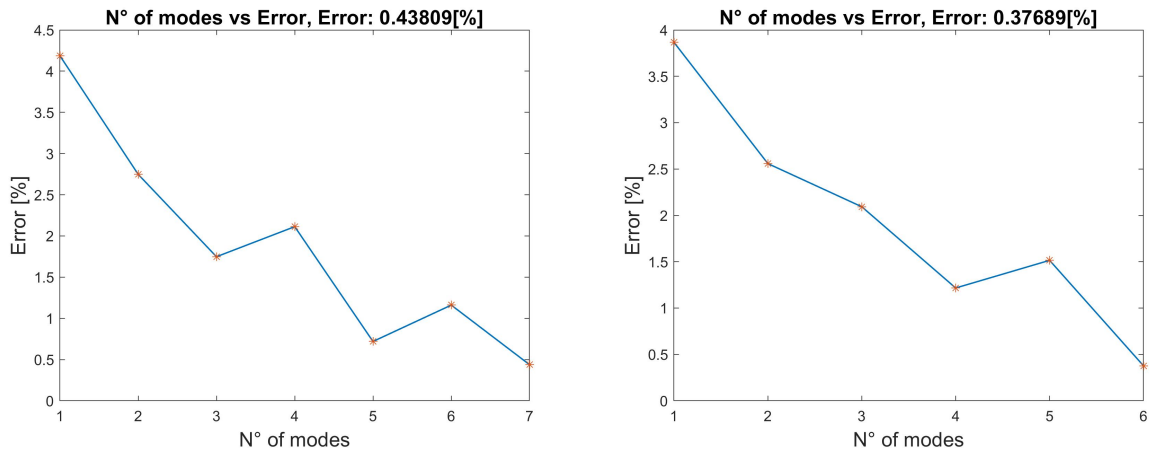


Figure 5.20: Damage evolution comparison between a classic and a multiscale temporal resolution approach.

The LATIN error versus the number of PGD modes obtained by the multiscale approximation is compared against the classic strategy in figure 5.21. An interesting point to be noticed is the difference in the number of modes needed to converge, showing that the convergence of the LATIN method is very sensitive on how the low-rank approximation is constructed.



(a) Classic LATIN-PGD

(b) Multiscale LATIN-PGD

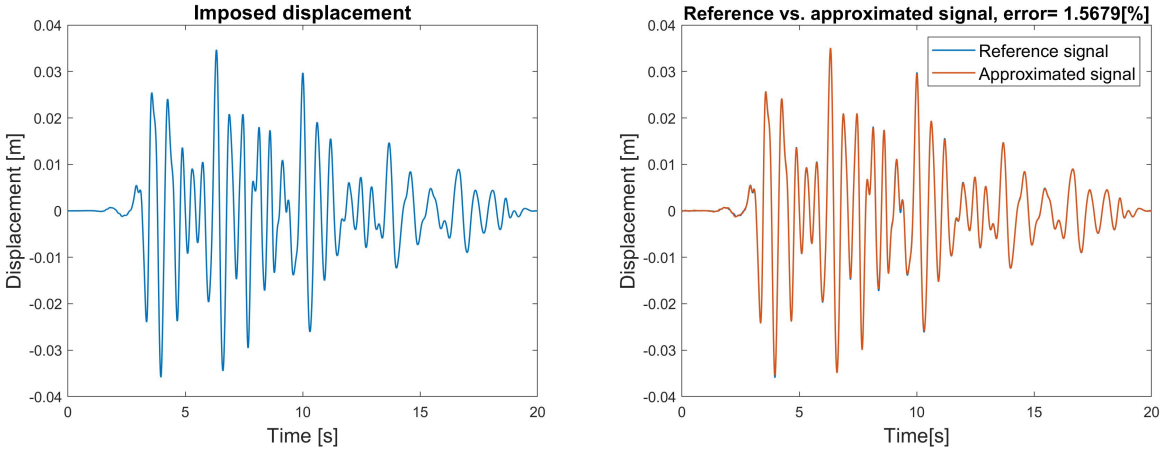
Figure 5.21: LATIN error vs number of PGD modes for the classic and multiscale temporal resolution.

Table 5.3 summarizes the computational times required for both strategies.

Solver	Computing time
Classic LATIN-PGD	2.3 minutes
Multiscale LATIN-PGD	2 minutes

Table 5.3: Time comparison for the case of isotropic damage.

The first example is the simplest when applying the multiscale approach, because the external excitation is monoperiodic. In order to carry out the method for a more general case, in the following we imposed to the structure a seismic signal which is shown in figure 5.22a.



(a) (b)

Figure 5.22: (a) Imposed displacement and (b) its approximation using the multiscale signal approach.

After the application of the signal approximation method presented in section 5.2, the signal of figure 5.22b is obtained, where the following 5 characteristic periods are obtained:

$$\tau = \left[0.24\ [s],\quad 0.36\ [s],\quad 0.52\ [s],\quad 0.68\ [s],\quad 0.92\ [s]\right] \tag{5.92}$$

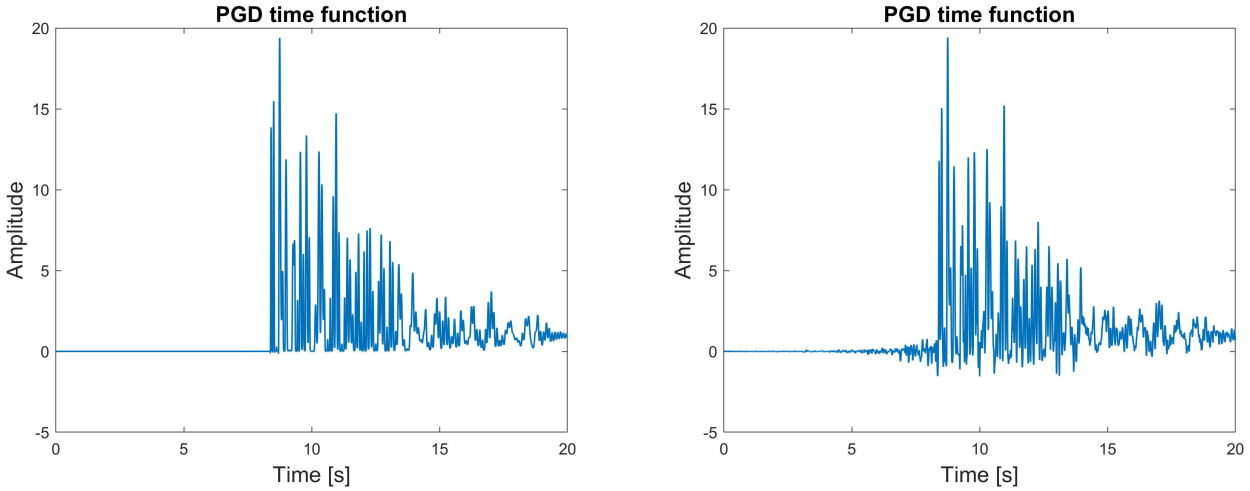
The temporal DOFs under this new situation are exposed below in table 5.4.

Solver	FEM DOFs	DOFs of micro problem (Sym + Asym)
Classic LATIN-PGD	4000	0
Multiscale LATIN-PGD	498	438

Table 5.4: Comparison of DOFs in time.

As can be seen in table 5.4, due to the larger number of characteristic periods obtained from the external excitation, the number of degrees of freedom associated with the multiscale approach increased considerably compared to the single-period case. In spite of this, the multiscale approach also allows to drastically reduce the DOFs to be determined compared to the classical method.

Figure 5.32 shows the temporal functions obtained for the first PGD mode when using the classic and the new multiscale strategy.



(a) Classic PGD time function.

(b) Multiscale PGD time function.

Figure 5.23: Comparison between the classic and the multiscale PGD approximation of the first temporal PGD function $\lambda_1(t)$.

From the figure above it can be seen that the multiscale method manages to correctly approximate the PGD time function in the case of a multi-periodic external excitation. As presented in previous sections, in the multi-periodic case the functions are approximated as the sum of sub-modes, in this sense the figure 5.24 shows the first 3 sub-modes obtained for the approximation of the function shown in figure 5.23b.

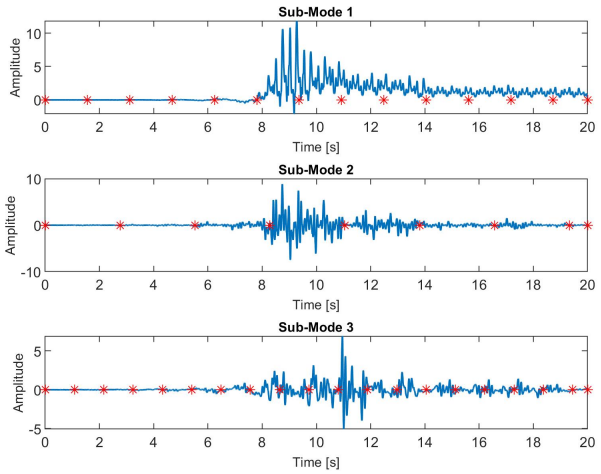


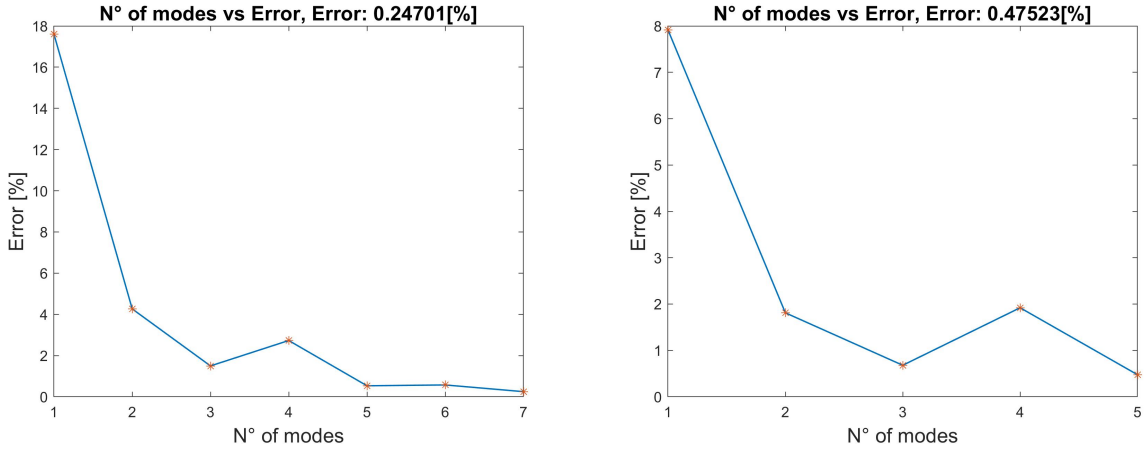
Figure 5.24: First 3 temporal multiscale sub-modes of the first temporal PGD function $\lambda_1(t)$.

The computational times needed for the resolution of the reference problem are summarized in table 5.5.

Solver	Computing time
Classic LATIN-PGD	5.5 minutes
Multiscale LATIN-PGD	5 minutes

Table 5.5: Time comparison for the case of isotropic damage.

Finally, figure 5.25 shown the curves of LATIN error versus the number of PGD modes for the classical and the multiscale approaches.



(a) Classic LATIN-PGD (b) Multiscale LATIN-PGD

Figure 5.25: LATIN error vs number of PGD modes for the classic and multiscale temporal resolution.

5.4.2 Elasto-visco-plasticity example

To illustrate the temporal multiscale approximation for the case of elasto-visco-plasticity, we reconsider the same monoperiodic external excitation as presented in the numerical example of section 4.6.2, but with a smoother increase of the amplitude to allow a better multiscale approximation:

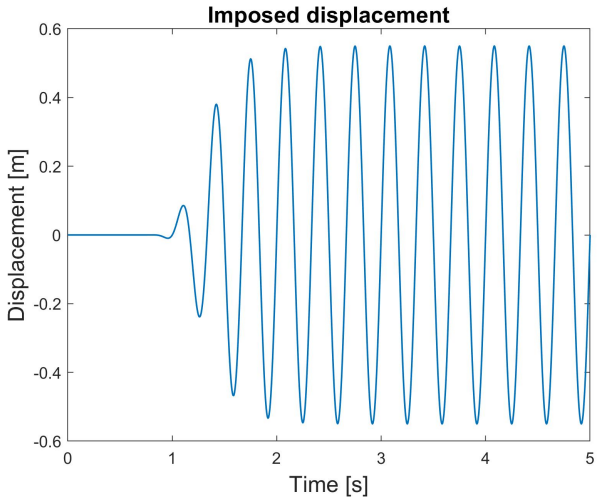
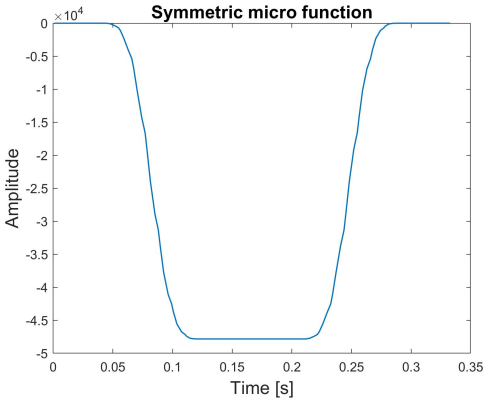
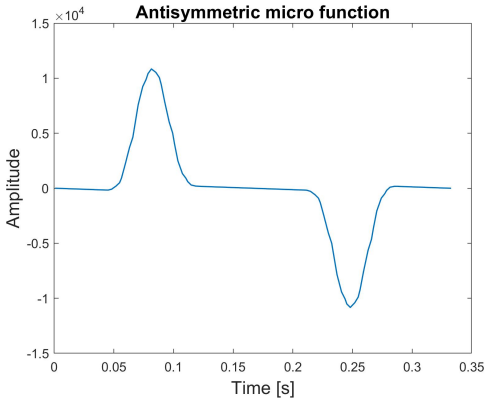


Figure 5.26: Imposed displacement.

This input signal have the same number of temporal DOFs as presented in table 5.2 for the case of isotropic damage and the same characteristic time ($\tau = 0.33\ [s]$). In this situation when applying the multiscale approach for the determination of the first temporal PGD function we obtain the following micro functions presented in figure 5.27. On the other hand the macro functions obtained are shown in figure 5.28.

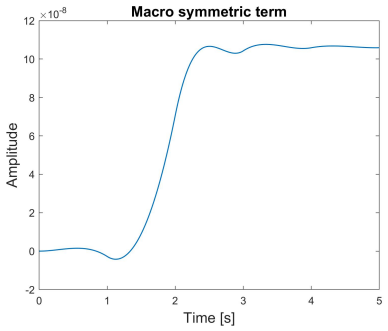


(a) Micro symmetric function.

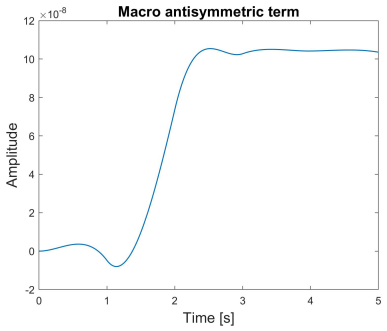


(b) Micro antisymmetric function.

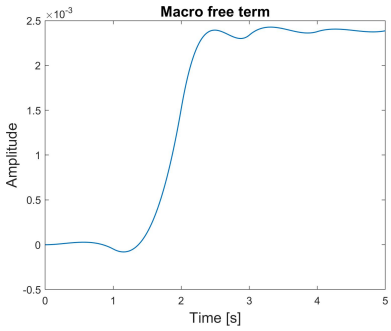
Figure 5.27: Micro functions.



(a) Macro symmetric term.



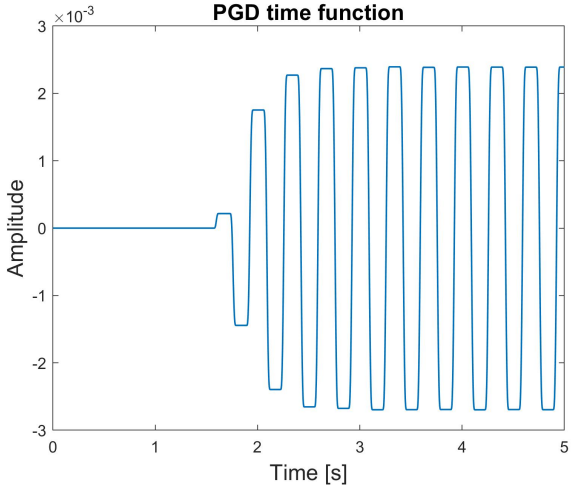
(b) Macro antisymmetric term.



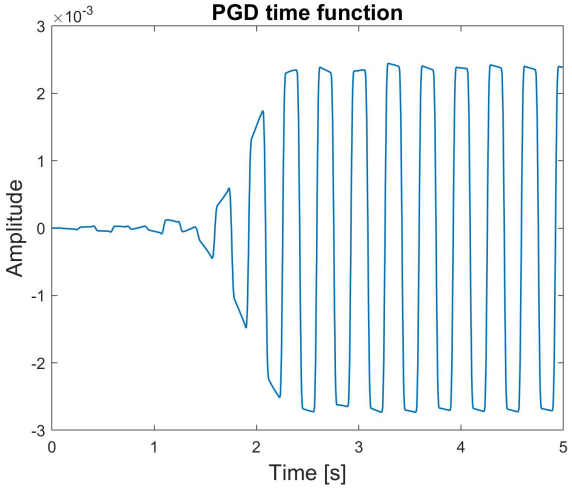
(c) Macro free term.

Figure 5.28: Macro functions.

Figure 5.29 shows a comparison between the first classical time PGD function and the multiscale one.



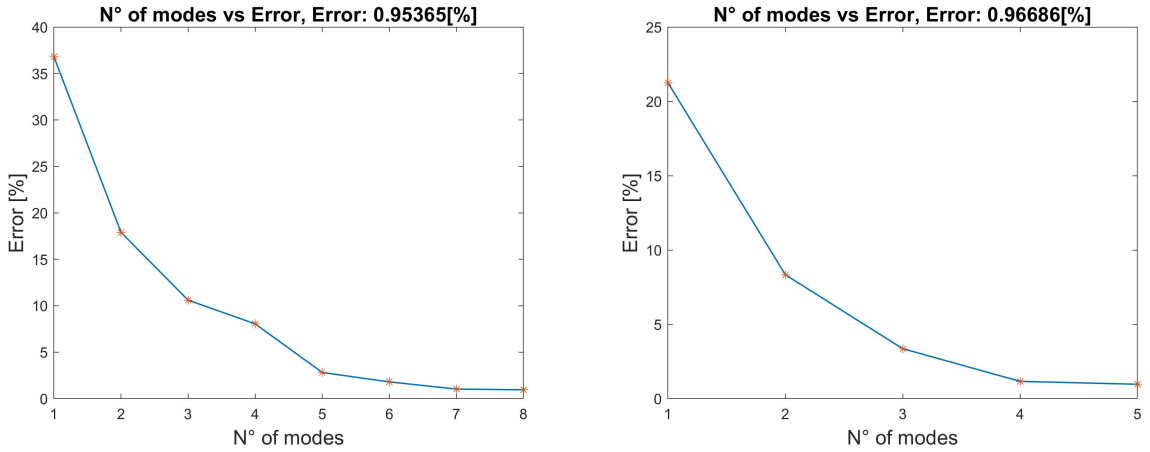
(a) Classic PGD time function.



(b) Multiscale PGD time function.

Figure 5.29: Comparison between the classic and the multiscale approximation of the first temporal PGD function $\lambda_1(t)$.

The multiscale approximation as for the case of isotropic damage have no difficulties in converging to the final solution, a comparison of the classic and multiscale LATIN error versus the number of PGD modes are shown in figure 5.30.



(a) Classic LATIN-PGD (b) Multiscale LATIN-PGD

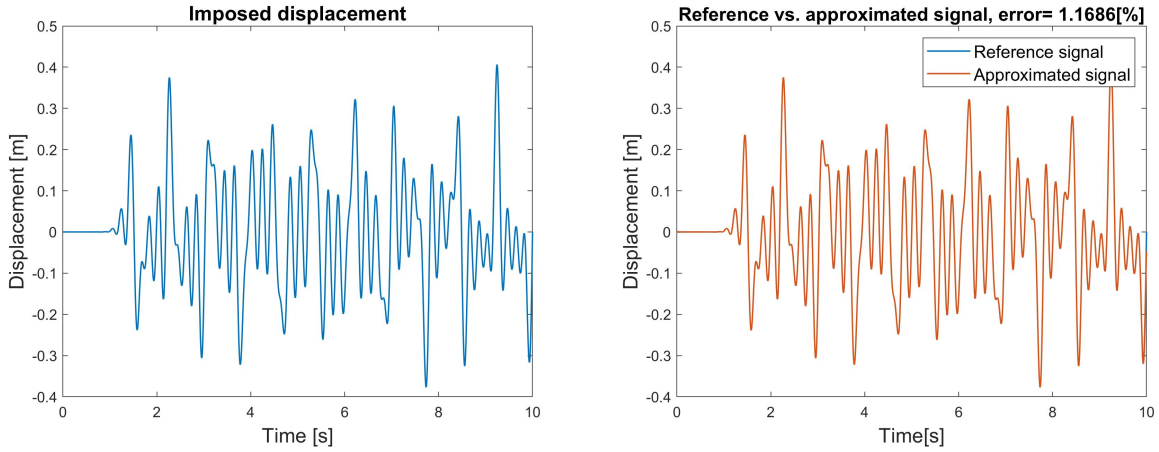
Figure 5.30: LATIN error vs number of PGD modes for the classic and multiscale temporal resolution.

Finally, the computational times for the resolution of the reference problem are given in table 5.6.

Solver	Computing time
Classic LATIN-PGD	2.8 minutes
Multiscale LATIN-PGD	2.1 minutes

Table 5.6: Time comparison for the case of elasto-visco-plasticity.

The temporal multiscale strategy is also applicable for a complex signal excitation. In the following we consider that the imposed displacement $u_z^D(t)$ is given by the signal shown in figure 5.31a, which is created artificially by the sum of several sinusoidal contributions.



(a) (b)

Figure 5.31: (a) Imposed displacement and (b) its approximation using the multiscale signal approach.

The characteristic period needed for the approximation of the signal 5.31a are given by:

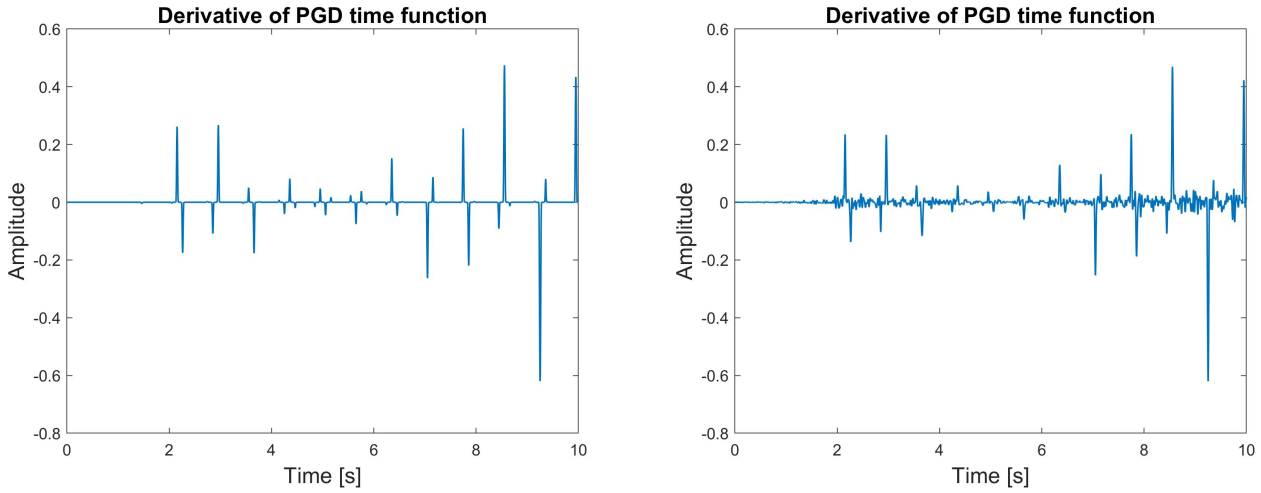
$$\tau = \left[0.13\ [s],\quad 0.2\ [s],\quad 0.24\ [s],\quad 0.27\ [s],\quad 0.39\ [s],\quad 0.68\ [s]\right] \tag{5.93}$$

Under this situation, the DOFs associated to each resolution strategy is presented in table 5.7.

Solver	FEM DOFs	DOFs of micro problem (Sym + Asym)
Classic LATIN-PGD	4000	0
Multiscale LATIN-PGD	540	468

Table 5.7: Comparison of DOFs in time.

The approximation of the first temporal PGD function related to the stress and plastic deformation can be seen in figure 5.32 where the derivative of the PGD function is presented against the classic resolution result.



(a) Derivative of classic PGD time function.

(b) Derivative of multiscale PGD time function.

Figure 5.32: Comparison between the classic and the multiscale approximation for $\dot{\lambda}_1(t)$.

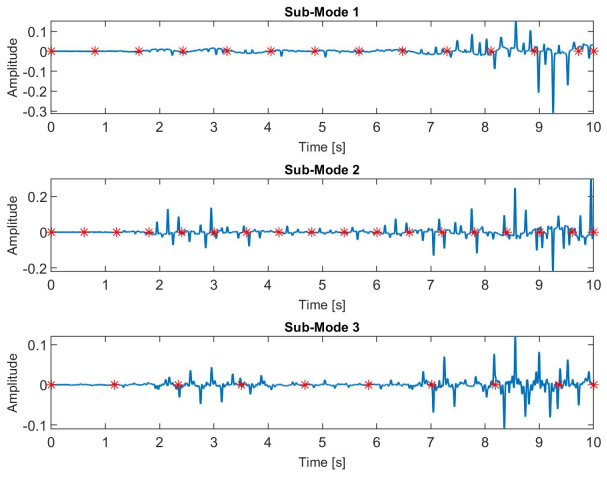
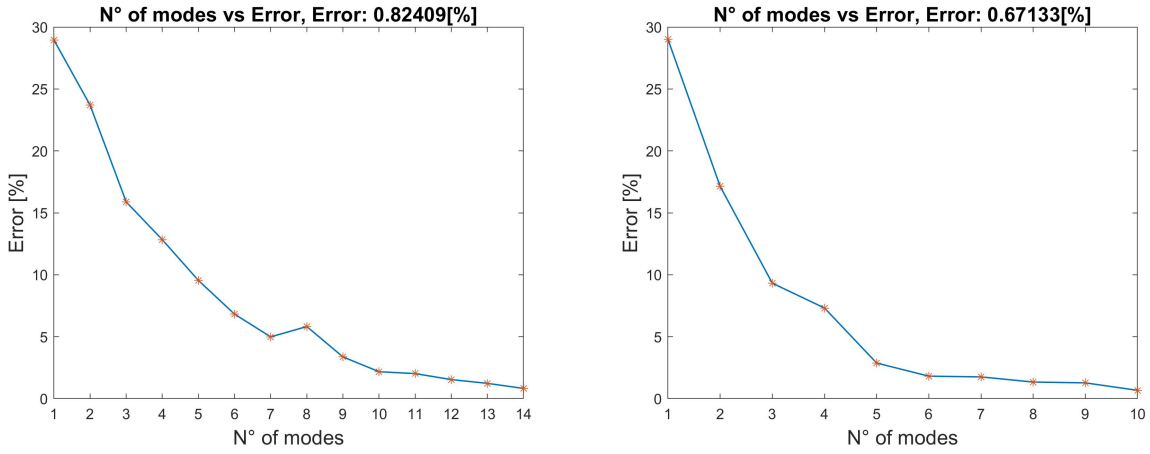


Figure 5.33: First 3 temporal multiscale sub-modes of $\dot{\lambda}_1(t)$.

In addition, figure 5.33 presents the first three derivative sub-modes of the approximation, where the red dots indicate the macro discretization.

Even if the approximation introduces some bifurcations or noise, the presented strategy fix those errors iteratively, therefore the convergence to the final solution is not affected, this can be verified by inspecting the curves of figure 5.34, which show the LATIN error versus the number of PGD modes required for both classic and multiscale approximations.



(a) Classic LATIN-PGD (b) Multiscale LATIN-PGD

Figure 5.34: LATIN error vs number of PGD modes for the classic and multiscale temporal resolution.

Finally, the computational times required for the resolution of the problem are resumed in table 5.8.

Solver	Computing time
Classic LATIN-PGD	10.18 minutes
Multiscale LATIN-PGD	10.1 minutes

Table 5.8: Time comparison for the case of elasto-visco-plasticity.

5.4.3 Conclusions on the numerical results

From the results presented in the previous section, two important points can be highlighted; (i) the drastic decrease in the temporal DOFs to be solved and (ii) the performance of the multiscale strategy, these points are detailed below.

(i) **Decrease of the temporal DOFs**:

In general, the number of DOFs needed to be solved decreases for all the numerical test performed, achieving a large reduction when a monoperiodic external excitation is considered for both the isotropic and the elasto-visco-plasticity case as shown in table 5.2 (both tests have the same TDGM and multiscale DOFs). Obtaining 1800 DOFs for the TDGM and only 60 for the multiscale approach (36 for the determination of the macro functions and 24 for the micro ones). However, the number of DOFs increases with the complexity of the input signal, which is related to the higher number of signal sub-modes needed to approximate the external excitation (see section 5.2). Of course, this increase in DOFs is directly related to the selected micro-macro ratio, which in the above examples has a fixed value of 3 for all the cases,

but other values could be used, modifying in this sense the number of DOFs. In this aspect, the new multiscale approach achieves very well the goal of reducing the temporal DOFs to be determined at each global stage of the LATIN-PGD method, overcoming one of the main limitation when solving large time duration problems in the context of the LATIN-PGD method.

(ii) **Performance of the strategy**:

Regarding computational times, the numerical tests performed for both isotropic damage and elasto-visco-plasticity allow a slight computational saving when using the multiscale approach. However, as can be seen from the curves of error versus number of modes for each case, the main reason is due to the smaller number of PGD modes needed to converge. The difference in the number of PGD modes in the convergence is due to the construction quality of the time functions, which affect the iterative processes such as the fixed-point technique in the enrichment step, as well as the stagnation of the updated time functions in the preliminary step, which consequently affect the convergence ratio of the LATIN-PGD method.

The computational savings related to temporal resolution using the multiscale approach is negligible compared to the TDGM, mainly due to the short temporal duration of all test cases. Moreover, as presented in chapter 4, the TDGM solves the temporal problem incrementally, which makes it a really efficient method. In contrast, the multiscale method (as developed in this chapter in order to improve the continuity of the approximation) uses a continuous approach in its macro FEM formulation, so its resolution is less efficient than a discontinuous approach and, therefore, reducing the computational time compared to the TDGM is really difficult. However, the comparison of the multiscale strategy with the TDGM was done in order to show that even a method as efficient as the TDGM is not more effective than the multiscale method, even when dealing with problems of relatively short duration.

Another interesting point of the temporal multiscale approximation is the number of LATIN iterations needed to converge. In general, due to the multiscale approximation of the temporal PGD functions, the number of iterations performed in the enrichment and preliminary steps are increased. This is the reason why, although fewer PGD modes are needed to converge when using the multiscale approximation, the computational times are very close to the classical LATIN-PGD using the TDGM. For instance, let us consider the complex input excitation for the elasto-visco-plasticity case, where the computational times are given in table 5.8. When using the multiscale approach the problem converges in 10 modes while the classical approach required 14, as shown in figure 5.34. However, the multiscale approach required 58 LATIN iterations to converge, while the classical approach required 59. The latter means that the multiscale approach required on average 6 iterations for each mode, while the classical approach required only 4. This phenomenon of increasing the number of LATIN iterations for each PGD mode calculated when using the multiscale method is repeated in all the numerical tests performed. This is a disadvantage, since in the view of solving problems in which the time domain is very large, an increase in the number of LATIN iterations would mean a considerable increase in the computational cost, due to the additional operations to be performed in both the local and global stages.

5.5 Conclusions

The LATIN method together with the PGD has managed to reduce the computational costs when solving nonlinear problems. Despite this advantage there are still certain problems that the LATIN-PGD cannot address due to its high computational cost, for instance, when the external excitation have a rich frequency content and it have a large duration in time like fatigue loads or seismic excitations. In these cases, the large temporal domain induce that a large number of temporal degrees of freedom must be determined at each global stage of the LATIN-PGD method, reducing in this way its performance.

In this context, a new temporal multiscale approximation has been introduced in this chapter with the objective of improving the efficiency of the temporal resolution at the global stage of the LATIN-PGD method. The main idea of this strategy consists in first determining some very important parameters called *characteristic periods*, that are obtained from the imposed external excitation which represent the periods of the different repeating patterns in the signal. These characteristic periods are determined by approximating the external excitation using a multiscale approach, which approximates the reference signal as the sum of functions with macro and micro behavior, where the micro functions correspond to sinusoids and the macro terms to polynomial functions that have a slow evolution in time. Once the characteristic periods are determined, these parameters are used to approximate the temporal PGD functions in the global stage of the LATIN method. The temporal PGD functions are also approximated using two time scales, one macro and one micro, but in this case the micro functions differ from the sinusoids and must be properly determined using the assumption that the different micro functions have a periodicity equals to the characteristic periods associated with the external excitation. Once the micro functions are determined, the associated macro terms are calculated. This new multiscale strategy was applied for two different constitutive relations, the isotropic damage for concrete materials and elasto-visco-plasticity for metals, where the difficulties and the mathematical framework associated with each of them were exposed.

This method allows to reduce the total number of DOFs associated with the time resolution when dealing with long duration input excitations, which is achieved due to the coarse discretization needed for the approximation of the macro terms and the few DOF needed for the determination of the micro functions. However, as discussed in section 5.4.3, the multiscale approximation tends to increase the LATIN iterations needed to compute each PGD mode on average, which is a disadvantage if long duration external excitations are considered where several PGD modes should be determined to reproduce the final solution. The latter is due to the additional local and global stages that need to be solved for the additional LATIN iterations, which in the context of large time domains translates into large operations for the constitutive relation evaluation in the local stage, as well as costly spatio-temporal integration operations in the global stage at the enrichment and preliminary steps.

It follows from the above that in order to really improve LATIN-PGD when dealing with large time domains two aspects need to be optimized. The first one consists in the improvement of the multiscale approach to reduce the supplementary LATIN iterations and secondly, dedicated strategies must be developed to decrease the cost of each LATIN iteration. The first point is left as a perspective for future developments, where different strategies can be included to the multiscale approach to improve its construction, such as a better construction of the micro functions, or the introduction of the TDGM formulation to obtain the macro terms. The second point, is discussed in the next chapter 6, where a hyper-reduction technique is considered to speed up the evaluation of the local stage and the spatio-temporal integrations in the global stage.

The new multiscale strategy still needs to be refined and improved. Nevertheless, the multiscale method presented in this chapter proves to be a robust method, allowing the multiscale approximation of complex responses over time for both isotropic damage and elasto-visco-plasticity. The ideas presented in this chapter are not only intended to deliver a solution to optimize the LATIN-PGD method, but also to provide a new framework for the treatment of nonlinear problems where the external excitation has a large time duration and complex behavior. Where these ideas can serve for future developments that could allow further optimization of the time resolution in the LATIN-PGD method.

Chapter 6

Hyper-reduction technique applied to the LATIN-PGD method

As shown in chapter 4, the LATIN-PGD allows a reduction in the computational expenses when solving low-frequency dynamics nonlinear problems. Additionally, chapter 5 adapted the temporal resolution at the global stage when dealing with large temporal excitations with complex behavior by introducing a new multiscale resolution. Both developments aim at optimizing the LATIN-PGD method, however, reducing only the cost of the global stage resolution is not enough to achieve a high-performance solver. Due to the global resolution strategy of the LATIN method, a big part of the solver's cost is indeed related to the evaluation of the constitutive relation and the integrals over the space and time needed for the calculation of the low-rank PGD decomposition. To reduce this expenses, this chapter introduces a hyper-reduction technique based on the Gappy POD principle to compute the constitutive relations on the entire space-time domain as a low-rank approximation. This approach allows not only a faster evaluation of the constitutive relations, but also faster iterations of the LATIN method, since the spatio-temporal integrals involved for the construction of the PGD approximation are simplified. The chapter ends with an application in elasto-visco-plasticity comparing the classical LATIN-PGD implementation with the new version including hyper-reduction.

Contents

6.1 Introduction **143**
6.2 Hyper-reduction technique for function approximation **145**
6.3 Hyper-reduced LATIN-PGD applied to elasto-visco-plasticity **150**
6.4 Numerical example **159**
6.5 Conclusions **161**

6.1 Introduction

As seen in previous chapters, the computational cost related to the global stage resolution is greatly reduced by using the PGD, however, the local stage remains unchanged, where a direct calculation of the constitutive relations is simply performed. This classical resolution strategy have a deep inconvenient when solving problems defined on large space-time domains. In fact, as exposed in [Capaldo, 2015], when the spatio-temporal domain becomes large, the numerical overhead of the LATIN-PGD increases drastically due to the amount of data to be handled and operated at each iteration (in addition to the cost related to the global stage resolution). The last is due to the fact that in each LATIN iteration a large number of evaluations at each integration point of the spatio-temporal domain must be performed to evaluate the constitutive relations at the local stage, as well as a large number of operations needed to perform the reduced basis projections and the spatio-temporal integrations required at the global stage for the determination of the PGD functions. To overcome one of these limitations, specifically the reduced basis projection costs, so-called hyper-reduction methods have been widely developed mainly in the context of nonlinear incremental solvers using model reduction techniques. Among these methods we can mention the EIM [Grepl et al., 2007] (used in the context of CRBM, see chapter 3) and its discretized version DEIM [Chaturantabut and Sorensen, 2010], or the ECSW method [Farhat et al., 2014, Farhat et al., 2015, Chapman et al., 2017, Farhat et al., 2020, Grimberg et al., 2021], where the result of the reduced basis projection on the discretized linearized equations is approximated using a coarse FEM mesh, thus reducing numerical operations and projection costs. The first hyper-reduction method applied to the LATIN-PGD corresponds to the *Reference Point Method* (RPM) presented in [Capaldo, 2015, Capaldo et al., 2017]. The RPM was developed specifically to reduce the number of operations in the LATIN-PGD when using non-constant tangent operators ($\mathbb{H}_\sigma$, $\hat{\mathbb{H}}$ and $H_{\bar{R}}$) when solving elasto-visco-plasticity problems via the approximation of these operators. The main idea of the RPM consists in the approximation of a given function over the space-time domain by considering a separate variable representation by space-time patches, whose functions are computed analytically and inexpensively. The patches are formed by considering a set of *reference points* in the spatial and temporal domain, which are used to construct the final approximation, that can be written as:

$\forall(i,j)\ \forall(\underline{x},t) \in \Omega_i \times I_j,$

$$f(\underline{x},t) \approx \bar{f}(\underline{x},t) = \bar{a}_{ij}(\underline{x})\bar{b}_{ij}(t) \tag{6.1}$$

with $\bar{a}_{ij}(\underline{x})$ and $\bar{b}_{ij}(t)$ the spatial and temporal functions of the approximation which are defined on a given spatio-temporal patch $\Omega_i \times I_j$, defined by the reference points as shown in figure 6.1.

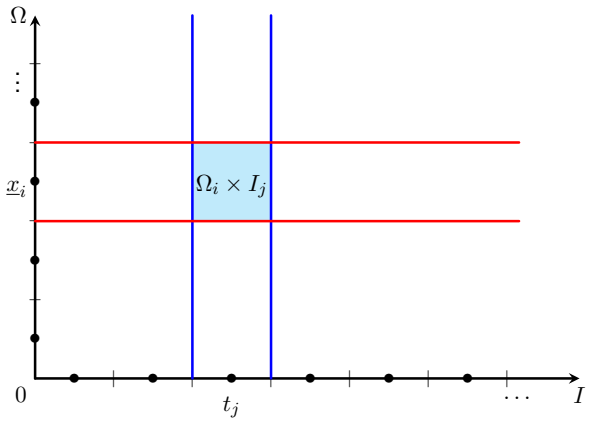
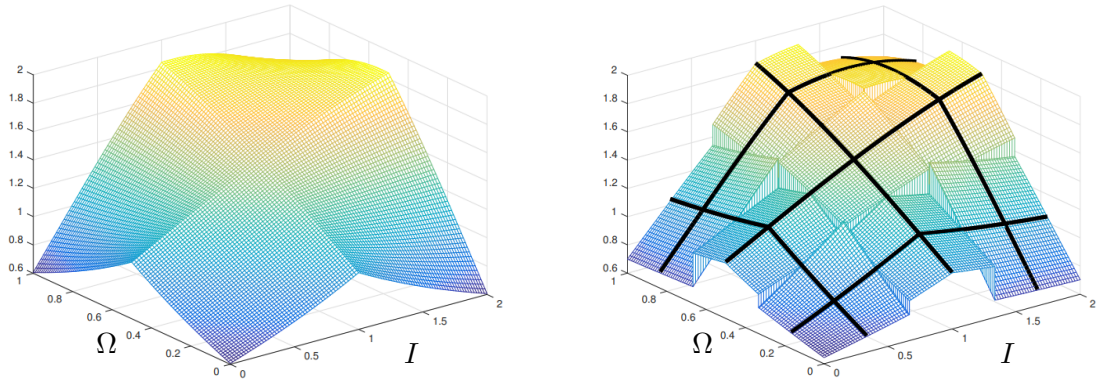


Figure 6.1: Coarse patches defined by the reference points ($\underline{x}_i$ and t_j) in space and time of the RPM.

Figure 6.2 show one of the results presented in [Capaldo et al., 2017] for the approximation of a given func-

tion by using the RPM. Figure 6.2a shows the reference space-time function and 6.2b its approximation by the RPM. It is important to note that the RPM representation is continuous only by patches.



(a) Reference function. (b) Approximated function by the RPM.

Figure 6.2: Reference function (left) and approximation by using the RPM (right).

The RPM allowed an optimized LATIN-PGD, where a faster convergence is achieved due to: (i) a faster construction of the tangent operators in the local stage and (ii) an optimized global stage due to the use of the tangent operators and the reduction of the projection operations in the global stage by using a separate variable approximation by patches of them. However, despite the contributions of the RPM, the calculation of the constitutive relations remained unchanged. The main reason for not approximating the constitutive relation with the RPM is the lack of continuity of the approximation generated, as can be seen in figure 6.2b. Since the LATIN method converges when the distance between the local and global solution is less than a given threshold, the lack of continuity when approximating the variables of the constitutive relations in the local stage would generate at convergence a solution with many associated errors.

In order to achieve a continuous approximation of the constitutive relations at the local stage to effectively optimize the LATIN-PGD method when dealing with long duration input excitations, a different approach is considered in the present chapter. For this purpose, a hyper-reduction technique based on the Gappy POD is used at the local stage of the LATIN solver. The main idea is to economically reconstruct the quantities of the constitutive relations over the totality of the integration points by means of a low-rank approximation, which can allow faster projective and integration operations at the global stage due to its separate variable representation. The construction of the low-rank approximation is performed by exploiting the same idea of the spatio-temporal reference points used in the RPM, which leads to an economic reconstruction. It should be noted that the tangent operators are not approximated here since in this thesis work they are chosen to be constant (see chapter 4). The Gappy POD was first introduced in [Everson and Sirovich, 1995] for facial image reconstruction and subsequently applied in the *A Priori Hyper Reduction* method [Ryckelynck, 2005, Ryckelynck et al., 2006a, Ryckelynck, 2009, Ryckelynck et al., 2011, Ryckelynck et al., 2012], in which the full-field solution is obtained by solving the problem on certain finite elements selected from the entire discretized domain, giving it its name “Hyper Reduction” (see section 3.3.3). The main difference between the ideas presented for face image reconstruction or the *A Priori Hyper Reduction* method and the strategy presented in this chapter is that this reconstruction is performed on the complete spatio-temporal domain due to the global nature of the LATIN solver.

In the following sections the hyper-reduction strategy is presented, where its effectiveness for the approximation of a 2D function is first demonstrated to better illustrate the main steps of the method. Finally the hyper-reduction strategy is applied to the LATIN-PGD method for the elasto-visco-plasticity case (the

reasons why it is applied to this behavior will be given in the following sections). This chapter ends with a comparison of the computational cost between the classical LATIN-PGD method (as presented in chapter 4) and the hyper-reduced LATIN-PGD method.

6.2 Hyper-reduction technique for function approximation

As mentioned earlier, the strategy considered to reduce the computational cost associated with the local step evaluation of the LATIN method is the Gappy POD. The idea of using this method for the approximation of the constitutive relations quantities is not new, in fact it has been applied in the context of the A Priori Hyper Reduction method [Ryckelynck, 2009], where a spatial POD basis is constructed from the known solution of the nonlinear constitutive relation in an offline procedure. This is done in order to make a quick evaluation of the material behavior for different time steps in an online resolution. However, since the quantities of interest are determined at a given time, their low-rank approximation is written in a general way as follows:

$\forall \underline{x} \in \Omega,\ t_j \in I,$

$$f(\underline{x}, t_j) \approx \sum_{i=1}^{m_r} \Theta_i(\underline{x}) a_{ij} \tag{6.2}$$

where $\Theta_i(\underline{x})$ correspond to the space reduced order basis functions and a_{ij} their associate scalar factors $\forall i \in [1, ..., m_r]$, with m_r denoting the rank of the approximation. The numeric value of the scalar factors a_{ij} must be calculated for each solution at a given time step t_j to be approximated, the spatial POD functions are known and determined beforehand in an offline stage.

Due to the global space-time nature of the LATIN method, the novelty of the approach that is proposed herein consists in no longer considering scalar factors a_{ij} associated to each space POD functions, but temporal functions itself $\mu_i(t)$:

$\forall \underline{x} \in \Omega,\ \forall t \in I,$

$$f(\underline{x}, t) \approx \sum_{i=1}^{m_r} \Theta_i(\underline{x}) \mu_i(t) \tag{6.3}$$

The main problem consists in calculating the functions $\Theta_i(\underline{x})$ and $\mu_i(t)$, $\forall i \in [1, ..., m_r]$ in an inexpensive way; to do so, we use the concept of reference points in space and time in order to construct this low-rank approximation using limited information of $f(\underline{x}, t)$ on these points.

In the following sections, the main steps of the considered hyper-reduction, i.e. the computation of the spatial and temporal functions, are presented. These sections introduce in a simple way the main ideas of the strategy that will be used later for the approximation of the constitutive relations quantities in the LATIN method.

6.2.1 Reference points in time: determination of the spatial POD functions

Lets consider a reference function $f(\underline{x}, t)$ defined over a domain $\Omega \times I$, and suppose that this function allows a low-rank approximation in the form of equation (6.3). This means that the original function evaluated on some reference points in time can be also approximated by this low-rank representation:

$$f(\underline{x}, \bar{t}) \approx \sum_{i=1}^{m_r} \bar{\Theta}_i(\underline{x}) \bar{\mu}_i(\bar{t}) \tag{6.4}$$

where $\bar{t} = \{t_1, t_2, ..., t_{\bar{n}_{t,g}}\}$ correspond to the reference points in time which are chosen following a regular discretization over the temporal domain $I = [0, T]$ as seen in figure 6.3.

Since the POD approximation is done over an incomplete set of temporal data points, the functions $\bar{\Theta}_i(\underline{x})$ and $\bar{\mu}_i(t)$ are a priori different from the real POD approximation of the complete function $f(\underline{x},t)$, however these functions remains a good approximation. By consequence, in what follows we suppose that the original space POD functions of $f(\underline{x},t)$ is well approximated by the spatial POD functions obtained by using the expression (6.4) only defined over few time reference points, this is:

$\forall i \in [1,...,m_r]$,

$$\Theta_i(\underline{x}) \approx \bar{\Theta}_i(\underline{x}) \tag{6.5}$$

where $\Theta_i(\underline{x})$ is the reference space POD function if the POD is applied to the whole space-time domain $\Omega \times I$. Therefore the spatial function of the whole hyper-reduced approximation is simply given by:

$\forall i \in [1,...,m_r]$,

$$\Theta_i(\underline{x}) = \bar{\Theta}_i(\underline{x}) \tag{6.6}$$

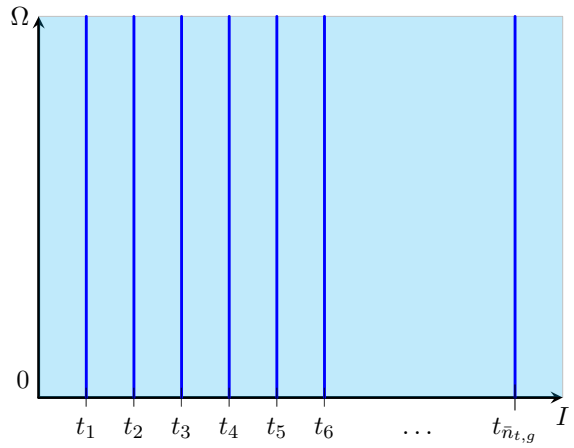


Figure 6.3: Reference integration points in time.

The functions $\bar{\Theta}_i(\underline{x})$ are simply obtained by following the procedure described in section 3.2.2.2 applied to the incomplete data set. That is, if we denote the discretized matrix that contains the values of the function at the spatial integration points (whose quantity is denoted as $n_{s,g}$) and at the temporal reference points $\bar{t}$ as $\underline{\underline{\boldsymbol{f}}} \in \mathbb{R}^{n_{s,g}} \otimes \mathbb{R}^{\bar{n}_{t,g}}$, the discretized spatial POD functions $\underline{\bar{\boldsymbol{\Theta}}}_i$ $\forall i[1,...,m_r]$ are determined by simply calculating the first m_r eigenvectors of the correlation matrix $\underline{\underline{\boldsymbol{C}}} \in \mathbb{R}^{n_{s,g}} \otimes \mathbb{R}^{n_{s,g}}$ given by:

$$\underline{\underline{\boldsymbol{C}}} = \underline{\underline{\boldsymbol{f}}}\,\underline{\underline{\boldsymbol{f}}}^T \tag{6.7}$$

Remark 1: The eingenvectors associated with the largest m_r eigenvalues of the correlation matrix of equation (6.7) directly give the discretized POD functions. However, the correlation matrix $\underline{\underline{\boldsymbol{C}}} = \underline{\underline{\boldsymbol{f}}}^T\underline{\underline{\boldsymbol{f}}}$ can also be considered, especially when the quantity of reference temporal points are less than the spatial integration points. In this situation the eigenvectors of this new correlation matrix will give temporal functions, which projected onto the matrix $\underline{\underline{\boldsymbol{C}}}$ finally give the discretized spatial POD functions (see section 3.2.2.2).

Remark 2: In the case where the function to approximate $\underline{\underline{f}}(\underline{x},t)$ is a tensor (as is the case when approximating the elasto-visco-plasticity quantities as will be seen in section 6.3.1), the spatial POD functions follow the same resolution strategy. In this case the discretized matrix $\underline{\underline{\boldsymbol{f}}}$ is formed by rearranging in the rows the values corresponding to the integration points in space (considering all the components of the tensor) and in the columns those associated to the temporal ones. Finding in this way, a problem equivalent to the one presented in this section, whose solution follows the same steps previously exposed.

6.2.2 Reference points in space: determination of the temporal POD functions

As mentioned above, the low-rank approximation to be determined is the one presented in the equation (6.3), the latter of which is recalled below:

$\forall \underline{x} \in \Omega,\ \forall t \in I,$

$$f(\underline{x}, t) \approx \sum_{i=1}^{m_r} \Theta_i(\underline{x}) \mu_i(t)$$

Since the last section, by using the reference points in time the spatial POD functions $\Theta_i(\underline{x})$ were determined, now, for the determination of the temporal functions $\mu_i(t)$ we simply employ the knowledge of the reference function $f(\underline{x}, t)$ over the reference points in space, minimizing the reconstruction error over those points, that is:

$\forall i \in [1, ..., m_r]\,,$

$$\{\mu_i(t)\} = \underset{\{\mu_i(t)\}_{i=1}^{m_r} \in \mathcal{U}^T}{\arg\min} \left|\left|\left| \sum_{i=1}^{m_r} \Theta_i(\bar{\underline{x}}) \mu_i(t) - f(\bar{\underline{x}}, t) \right|\right|\right|_2^2 \tag{6.8}$$

with:

$$|||\cdot|||_2^2 = \int_{\Omega_r \times I} (\cdot)^2 \, d\Omega dt$$

where $\bar{\underline{x}} = \{\underline{x}_1, \underline{x}_2, ..., \underline{x}_{\bar{n}_{s,g}}\}$ correspond to the group of reference points in space and Ω_r to the reduced domain defined by these reference points selected as shown in figure 6.4.

Figure 6.4: Reference Gauss points in space.

By minimizing the problem of equation (6.8) we obtain:

$\forall \delta\mu_i(t) \in \mathcal{U}^T,$

$$\sum_{i=1}^{m_r} \sum_{j=1}^{m_r} \int_I \delta\mu_i(t) \mu_j(t) A_{ij}^r dt = \sum_{i=1}^{m_r} \int_I \delta\mu_i(t) D_i^r(t) dt \tag{6.9}$$

with the constants and temporal functions given by:

$\forall (i, j) \in [1, ..., m_r]\,,$

$$A_{ij}^r = \int_{\Omega_r} \Theta_i(\bar{\underline{x}}) \Theta_j(\bar{\underline{x}}) d\bar{x} \tag{6.10}$$

$$D_i^r(t) = \int_{\Omega_r} \Theta_i(\bar{\underline{x}}) f(\bar{\underline{x}}, t) d\bar{x} \tag{6.11}$$

The terms of equations (6.10) and (6.11) are simply approximated by the following operations on discretized vector fields $\underline{\boldsymbol{\Theta}}(\bar{\underline{x}})$ and $\underline{\boldsymbol{f}}(\bar{\underline{x}}, t)$ as:

$\forall (i,j) \in [1,...,m_r]$,

$$A^r_{ij} = \underline{\boldsymbol{\Theta}}^T_i(\bar{\underline{x}})\,\underline{\boldsymbol{\Theta}}_j(\bar{\underline{x}}) \qquad (6.12) \qquad D^r_i(t) = \underline{\boldsymbol{\Theta}}^T_i(\bar{\underline{x}})\,\underline{\boldsymbol{f}}(\bar{\underline{x}},t) \qquad (6.13)$$

with the discretized vectors over the reference points in space given by:

$\forall i \in [1,...,m_r]$,

$$\underline{\boldsymbol{\Theta}}_i(\bar{\underline{x}}) = \left[\Theta_i(\underline{x}_1),\cdots,\Theta_i(\underline{x}_{\bar{n}_{s,g}})\right]^T \quad , \quad \underline{\boldsymbol{f}}(\bar{\underline{x}},t) = \left[f(\underline{x}_1,t),\cdots,f(\underline{x}_{\bar{n}_{s,g}},t)\right]^T$$

For the case where the functions $\Theta_i(\bar{\underline{x}})$ and $f(\bar{\underline{x}},t)$ are tensors, the above discretized operations remains unchanged by a simple reshape of the discretized multi-dimensional matrix into a vector form. Since the choice of the test functions $\delta\mu_i(t)$ is arbitrary, it can be simply chosen as non-zero values, this is, $\delta\mu_i(t) \neq 0,\ \forall i \in [1,...,m_r]$. With this choice, the resolution of the temporal POD functions is given by:

$$\begin{bmatrix} \mu_1(t) \\ \mu_2(t) \\ \vdots \\ \mu_{m_r}(t) \end{bmatrix} = \begin{bmatrix} A^r_{11} & A^r_{12} & \cdots & A^r_{m_r m_r} \\ A^r_{21} & A^r_{22} & & \\ \vdots & & \ddots & \\ A^r_{m_r 1} & & & A^r_{m_r m_r} \end{bmatrix}^{-1} \begin{bmatrix} D^r_1(t) \\ D^r_2(t) \\ \vdots \\ D^r_{m_r}(t) \end{bmatrix} \qquad (6.14)$$

The above expression has many numerical advantages, one of which is the inversion of a small constant matrix of size $m_r \times m_r$, which is used to determine the time functions of the decomposition by performing simple matrix-vector multiplication operations, which can be performed completely in parallel, thus reducing the computational cost for the determination of the hyper-reduced approximation.

6.2.3 Synthesis of the strategy

The hyper-reduction strategy presented in this section consists in computing a low-rank approximation of a given function $f(\underline{x},t)$, this is:

$$f(\underline{x},t) \approx \sum_{i=1}^{m_r} \Theta_i(\underline{x})\mu_i(t) \qquad (6.15)$$

The low-rank representation is economically computed by exploiting the use of reference points in space and time, which allows to build the approximation on a compressed representation of the original function by applying the POD to these limited data (Gappy POD). This strategy can be summarized in two main steps, which are listed below:

- **Determination of the spatial functions** $\Theta_i(\underline{x})$: By using *reference points in time*, a compressed representation of the reference function $f(\underline{x},t)$ is obtained, written as $f(\underline{x},\bar{t})$ with the reference temporal points $\bar{t} = \{t_1,t_2,...,t_{\bar{n}_{t,g}}\}$. This representation contains the whole spatial information over some specific reference points in time chosen uniformly. With this data the spatial functions of the POD decomposition $\Theta_i(\underline{x}),\ \forall i \in [1,...,m_r]$ are determined following the steps given in section 6.2.1.
- **Determination of the temporal functions** $\mu_i(t)$: Once the spatial POD functions are determined, we exploit the use of *spatial reference points* in order to obtain another compressed representation of $f(\underline{x},t)$, written as $f(\bar{\underline{x}},t)$ with the reference spatial points $\bar{\underline{x}} = \{\underline{x}_1,\underline{x}_2,...,\underline{x}_{\bar{n}_{s,g}}\}$. This representation contains the whole information in time at given reference spatial points. In this sense, the temporal functions of the POD approximation are determined such that the low-rank decomposition (6.15) reconstructs this compressed representation as presented in section 6.2.2.

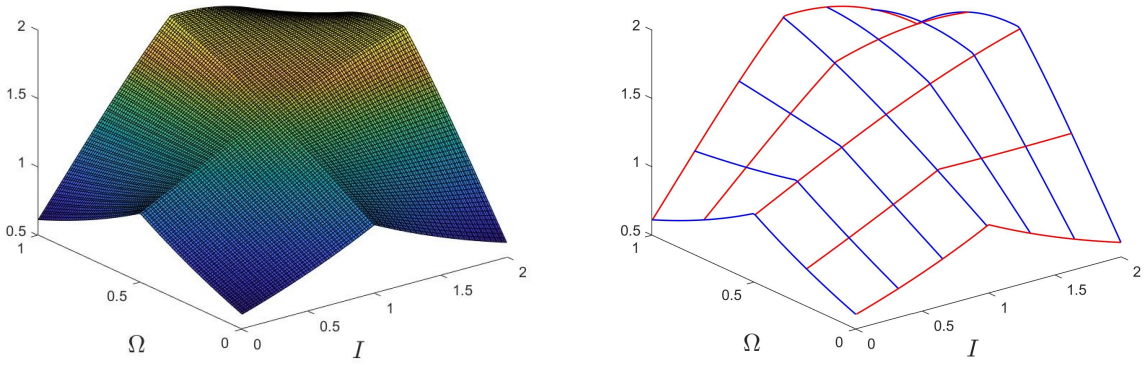
To illustrate the performance of the introduced hyper-reduced strategy, a numerical application consisting in the approximation of a given function is presented in the next section.

6.2.4 Example: 2D space-time function approximation

In the present section the hyper-reduced strategy is applied for the approximation of a 2D function which is presented in figure 6.5a and is given explicitly by:

$$f(x,t) = e^{-|(x-0.5)(t-1)|} + \sin(xt) \tag{6.16}$$

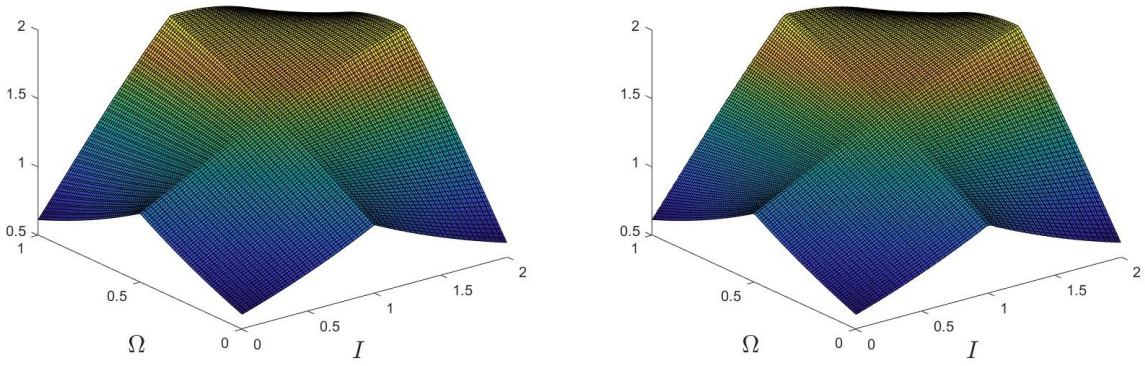
For the representation of the function, 1000 points in space and 3000 in time were considered in order to simulate a behavioral relationship where the time interval is large. On the other hand, for the determination of the approximation only 7 reference points in time and 5 in space were used as shown in figure 6.5b.



(a) Reference function. (b) Reference points.

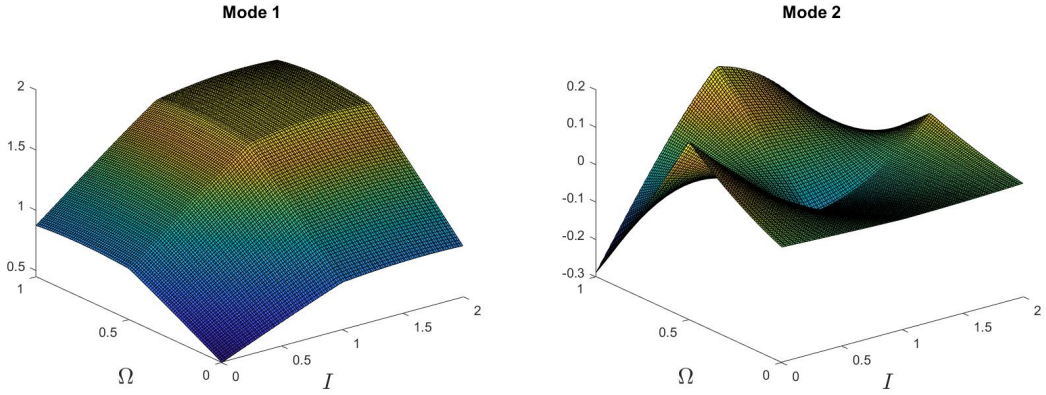
Figure 6.5: Reference function (left) and the function evaluated on the reference space-time points (right).

After the determination of the space and time POD functions as shown in sections 6.2.1 and 6.2.2 we obtained an approximation with an error of $0.01[\%]$ only by computing 3 POD modes, this approximation is compared against the reference space-time function in figure 6.6 and the POD modes obtained are shown in figure 6.7.



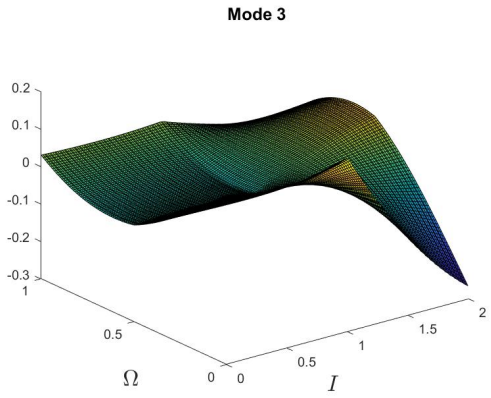
(a) Reference function. (b) Approximated function by using 3 POD modes.

Figure 6.6: Reference function and its approximation by the hyper-reduced strategy.



(a) First POD mode.

(b) Second POD mode.



(c) Third POD mode.

Figure 6.7: First 3 POD modes calculated for the approximation of the reference function.

The time required for the complete direct evaluation of the function takes a total of 0.0363 milliseconds while the low-rank approximation takes only 0.0017 milliseconds, which translates into a 95.23 $[\%]$ reduction in computational time.

Although the direct evaluation of the function in this particular case does not take much time (because it is an academic problem), the reduction in computational cost achieved with the hyper-reduction method is considerable and therefore its application to the evaluation of the local stage of the LATIN method naturally emerges as a viable option to reduce the computational time of the solver. However, it should be noted that the method can only be applied if the constitutive relation to be approximated can be evaluated explicitly in space and time. These types of constitutive relations fall into the so-called internal variable relationships [Ladevèze, 1999]. In this sense, the following section applies the presented strategy to the LATIN-PGD method, where an elasto-visco-plasticity constitutive relation is considered.

6.3 Hyper-reduced LATIN-PGD applied to elasto-visco-plasticity

In this section, the LATIN-PGD method is presented considering a hyper-reduction method applied to the local stage. As presented in previous chapters, the LATIN method solves the nonlinear problem over the whole space-time domain at each iteration of the solver. The main ingredients involved in the iterative resolution consist in defining a local and a global manifolds ($\boldsymbol{\Gamma}$ and $\boldsymbol{A_d}$ respectively), where the local one corresponds to the evaluation of the constitutive relations of the considered material and the global one

to the verification of the equilibrium and compatibility equations. The main idea consists in solving the reference problem at the global ($\mathcal{S} \in \boldsymbol{A_d}$) and local stages ($\hat{\mathcal{S}} \in \boldsymbol{\Gamma}$) in an iterative way by using the so called search directions, searching at each iteration of the method to decrease the distance between both solutions manifolds. This iterative procedure is remembered below:

$$\mathcal{S}_0 \in \boldsymbol{A_d} \longrightarrow \hat{\mathcal{S}}_{1/2} \in \boldsymbol{\Gamma} ... \longrightarrow \hat{\mathcal{S}}_{n+1/2} \in \boldsymbol{\Gamma} \longrightarrow \mathcal{S}_{n+1} \in \boldsymbol{A_d} ... \longrightarrow \bar{\mathcal{S}} \in \boldsymbol{A_d} \cap \boldsymbol{\Gamma}. \tag{6.17}$$

Being $\mathcal{S}_0$ the elastic solution calculated in dynamics. This process is ilustrated in figure 6.8 for the case of elasto-visco-plasticity.

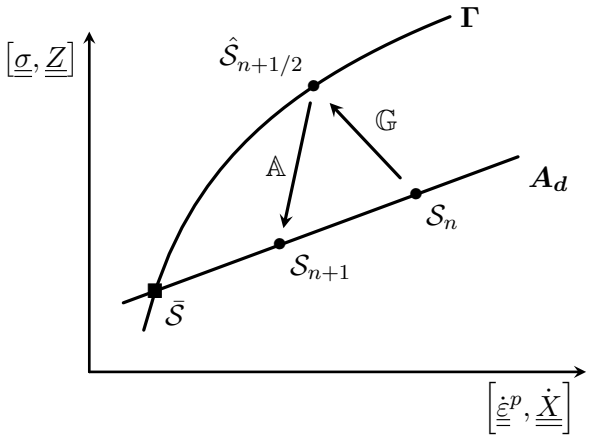


Figure 6.8: General resolution scheme of the LATIN solver.

In what follows, only the key and different points of the hyper-reduced LATIN-PGD method will be presented, especially the ones related to the low-rank approximation evaluation of the constitutive relations and the simplifications related to projections and integrals that must be done at the global stage. For more details about the different steps involved in the LATIN-PGD method for the case of elasto-visco-plasticity see chapter 4.

6.3.1 Local stage: hyper-reduced evaluation of the constitutive relations

After each global stage solution is determined ($\mathcal{S}_n \in \boldsymbol{A_d}$), the local stage solution must be computed ($\hat{\mathcal{S}}_{n+1/2} \in \boldsymbol{\Gamma}$), in where the nonlinear constitutive relations quantities are determined. In this section, the hyper-reduction method presented in section 6.2 is applied to the constitutive relation for its fast evaluation and its low-rank approximation, which will allow a more optimized global stage by speeding up the spatio-temporal integral operations to be performed as will be seen in section 6.3.2. The constitutive relations studied herein correspond to the elastic-visco-plasticity introduced in chapter 2 (see section 2.3.2). This choice is not arbitrary, in fact for the construction of the low-rank approximation we need to exploit the idea of reference integration points in space and time, which makes it necessary that the constitutive relations can be computed locally in space and time (each integration point is computed independently of the others). This is precisely the case of elasto-visco-plasticity, however this is not the case of isotropic damage, where the damage variable at a given time t^* depends on the solution of the damage for all time $t < t^*$ (see section 2.3.1), in where a simple ordinary differential equation (ODE) in time must be solved to find the temporal evolution of damage. For the particular case of isotropic damage or other types of constitutive relations that don't fit the requisite of being explicitly evaluated, further developed must be taken into account, such development are proposed as perspectives in section 6.5.

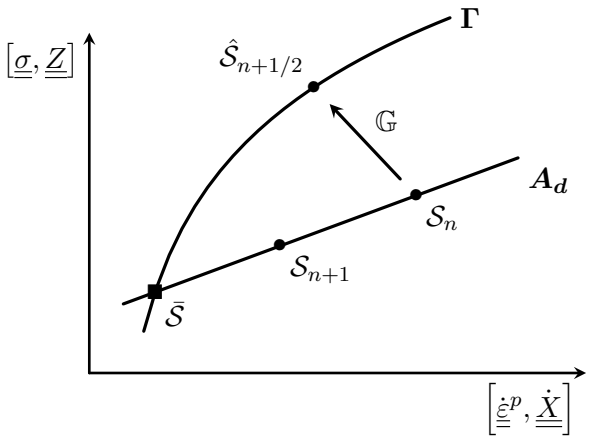


Figure 6.9: Calculation of the local stage solution $\hat{\mathcal{S}}_{n+1/2} \in \mathbf{\Gamma}$.

As presented in chapter 4, the variables that must be solved at the local stage when considering elasto-visco-plasticity are given by:

$$\hat{\mathcal{S}}_{n+1/2} = \{\hat{\dot{\underline{\underline{\varepsilon}}}}^p_{n+1/2}, \hat{\underline{\underline{\sigma}}}_{n+1/2}, \hat{\underline{\underline{\beta}}}_{n+1/2}, \hat{\dot{\underline{\underline{\alpha}}}}_{n+1/2}, \hat{\dot{\bar{r}}}_{n+1/2}, \hat{\bar{R}}_{n+1/2}\} \tag{6.18}$$

They are determined by solving the ascent search direction as illustrated in figure 6.9, which for the case of elasto-visco-plasticity is given by:

$$\mathbb{G}: \quad \left\{ \begin{bmatrix} \hat{\dot{\underline{\underline{\varepsilon}}}}^p_{n+1/2} - \dot{\underline{\underline{\varepsilon}}}^p_n \\ -(\hat{\dot{\underline{X}}}_{n+1/2} - \dot{\underline{X}}_n) \end{bmatrix} \right\} + \hat{\mathbb{H}}: \begin{bmatrix} \hat{\underline{\underline{\sigma}}}_{n+1/2} - \underline{\underline{\sigma}}_n \\ \hat{\underline{Z}}_{n+1/2} - \underline{Z}_n \end{bmatrix} = 0 \tag{6.19}$$

where the global stage variables at iteration n of the LATIN method are known and given by:

$$\mathcal{S}_n = \{\underline{u}_n, \underline{\underline{\varepsilon}}_n, \underline{\underline{\varepsilon}}^p_n, \underline{\underline{\sigma}}_n, \underline{\underline{\beta}}_n, \underline{\underline{\alpha}}_n, \bar{r}_n, \bar{R}_n\} \tag{6.20}$$

A constant ascent direction is used in order to simplify the computations at the local stage, this means that $\hat{\mathbb{H}}^{-1} = 0$, which means:

$$\begin{aligned} \hat{\underline{\underline{\sigma}}}_{n+1/2} &= \underline{\underline{\sigma}}_n \\ \hat{\underline{Z}}_{n+1/2} &= \underline{Z}_n = \left[\underline{\underline{\beta}}_n, \bar{R}_n\right] \end{aligned} \tag{6.21}$$

and therefore the remaining local variables are given as:

$$\hat{\dot{\underline{\underline{\varepsilon}}}}^p_{n+1/2}(\underline{x}, t) = k\langle f_s \rangle^{n_s}_+ \left[\frac{3}{2} \frac{\hat{\underline{\underline{\tau}}}_{n+1/2}}{\sqrt{\frac{3}{2}\hat{\underline{\underline{\tau}}}_{n+1/2} : \hat{\underline{\underline{\tau}}}_{n+1/2}}} \right] \tag{6.22}$$

$$\hat{\dot{\underline{\underline{\alpha}}}}_{n+1/2}(\underline{x}, t) = -k\langle f_s \rangle^{n_s}_+ \left[-\frac{3}{2} \frac{\hat{\underline{\underline{\tau}}}_{n+1/2}}{\sqrt{\frac{3}{2}\hat{\underline{\underline{\tau}}}_{n+1/2} : \hat{\underline{\underline{\tau}}}_{n+1/2}}} + \frac{a}{C} \hat{\underline{\underline{\beta}}}_{n+1/2} \right] \tag{6.23}$$

$$\hat{\dot{\bar{r}}}_{n+1/2}(\underline{x}, t) = -k\langle f_s \rangle^{n_s}_+ \left(\frac{\hat{\bar{R}}_{n+1/2}}{R_\infty} \frac{b}{2} - b^{\frac{1}{2}} \right) \tag{6.24}$$

with the plastic threshold function given by:

$$f_s = \sqrt{\frac{3}{2}\hat{\underline{\underline{\tau}}}_{n+1/2} : \hat{\underline{\underline{\tau}}}_{n+1/2}} + \frac{a}{2C} \hat{\underline{\underline{\beta}}}_{n+1/2} : \hat{\underline{\underline{\beta}}}_{n+1/2} - R_\infty \left(\frac{\hat{\bar{R}}_{n+1/2}}{R_\infty} \frac{b^{\frac{1}{2}}}{2} \right) \left(2 - \frac{\hat{\bar{R}}_{n+1/2}}{R_\infty} \frac{b^{\frac{1}{2}}}{2} \right) - \sigma_y \tag{6.25}$$

where $\hat{\underline{\underline{\tau}}}_{n+1/2} = \hat{\underline{\underline{\sigma}}}^D_{n+1/2} - \hat{\underline{\underline{\beta}}}_{n+1/2}$, with $\hat{\underline{\underline{\sigma}}}^D_{n+1/2}$ the deviatoric stress tensor.

Lets notice that the whole constitutive relation is given explicitly in space and time as required for the application of the hyper-reduction method.

Before starting with the following sections, it is necessary to define the space-time domain where the reference points will be selected in order to construct the low-rank approximation as presented in section 6.2.1. The data that we want to approximate consist of the plastic deformation, and the internal variables ($\hat{\underline{\underline{\varepsilon}}}^p_{n+1/2}$, $\hat{\underline{\underline{\alpha}}}_{n+1/2}$ and $\hat{\bar{r}}_{n+1/2}$), these variables are only defined on a limited space domain that we will define herein as Ω_{nl} such that $\Omega_{nl} \subset \Omega$ and over the whole time domain $I = [0, T]$. An example of a restricted spatial domain where plasticity occurs can be visualized in figure 6.10, where the red points indicate the presence of the cumulative plastic deformation for a bending test of a 3D beam.

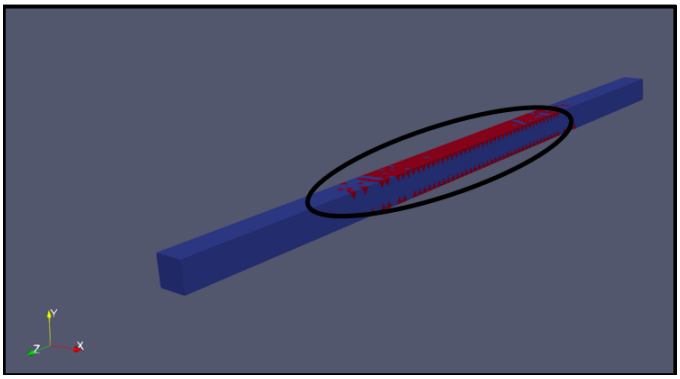


Figure 6.10: Localized nonlinear affected zone Ω_{nl}.

For the definition of Ω_{nl} we must first search all the elements and integration points in space that undergoes an elasto-visco-plastic behavior, this is simply done as follows:

$$\Omega_{nl} = \{\underline{x} \in \Omega \mid t \leqslant T,\ f_s(\underline{x}, t) > 0\} \tag{6.26}$$

with f_s the plasticity threshold function given in equation (6.25). The nonlinear variables appears only when the threshold function is positive, which give us a good indicator to determine the domain $\Omega_{nl} \subset \Omega$.

The idea behind the determination of the restricted domain Ω_{nl} consists in avoiding a big number of POD modes for the approximation of the local quantities. If some element on this restricted domain have a null value on the whole space-time domain, the required number of POD modes will increase since many recombination of them will be required to ensure this null value. In this sense, applying the hyper-reduced approximation to the restricted domain where all its elements undergo nonlinear behavior ensures a low number of POD modes.

The following sections present the main steps for the application of the hyper-reduction method to the approximation of the local stage quantities, which are the inexpensive construction of the space-time POD functions as presented for the simple example in section 6.2.

6.3.1.1 Reference points in time: determination of the spatial POD functions

The reference points in time are chosen equidistant on the whole interval $I = [0, T]$ as presented in figure 6.3. Of course a better choice of the most important points in time can be done by employing some optimization algorithm, but this is given as a perspective and not treated herein. The number of reference points in time are simply chosen as a fraction of the whole temporal integration points in I, this is:

$$\bar{n}_{t,g} = a_t n_{t,g} \tag{6.27}$$

where $n_{t,g}$ correspond to the total integration points in time and a_t the fraction number. The value of this fraction is based on empirical numerical tests and in the present work for the approximation of elasto-visco-plasticity is chosen to be $a_t = 0.2$.

By following section 6.2.1, the spatial POD functions of the hyper-reduced approximation of all the local stage quantities are determined such that the scattered data of the involved variables are well approximated by a low-rank POD decomposition, this is:

$\forall \underline{x} \in \Omega_{nl},\ \bar{t} \in I,$

$$\begin{aligned}
\hat{\underline{\underline{\varepsilon}}}^p_{n+1/2}(\underline{x},\bar{t}) &\approx \sum_{i=1}^{m_r^p} \bar{\underline{\underline{\Theta}}}^p_i(\underline{x})\bar{\mu}^p_i(\bar{t}) \\
\hat{\underline{\underline{\alpha}}}_{n+1/2}(\underline{x},\bar{t}) &\approx \sum_{i=1}^{m_r^\alpha} \bar{\underline{\underline{\Theta}}}^\alpha_i(\underline{x})\bar{\mu}^\alpha_i(\bar{t}) \\
\hat{\bar{r}}_{n+1/2}(\underline{x},\bar{t}) &\approx \sum_{i=1}^{m_r^{\bar{r}}} \bar{\Theta}^{\bar{r}}_i(\underline{x})\bar{\mu}^{\bar{r}}_i(\bar{t})
\end{aligned} \tag{6.28}$$

Where $\bar{t} = \{t_1, t_2, ..., t_{\bar{n}_{t,g}}\}$ correspond to the vector that contains all the reference points in time, and m_r^p, m_r^α and $m_r^{\bar{r}}$ to the rank of the approximation of the plastic deformation, kinematic hardening and isotropic hardening respectively. Following section 6.2.1, the spatial POD functions of the hyper-reduced approximation are chosen equal to those obtained in the previous process, that is:

$\forall i_1 \in [1, ..., m_r^p],\ \forall i_2 \in [1, ..., m_r^\alpha],\ \forall i_3 \in [1, ..., m_r^{\bar{r}}],\ \forall \underline{x} \in \Omega_{nl},$

$$\begin{aligned}
\underline{\underline{\Theta}}^p_{i_1}(\underline{x}) &= \bar{\underline{\underline{\Theta}}}^p_{i_1}(\underline{x}) \\
\underline{\underline{\Theta}}^\alpha_{i_2}(\underline{x}) &= \bar{\underline{\underline{\Theta}}}^\alpha_{i_2}(\underline{x}) \\
\Theta^{\bar{r}}_{i_3}(\underline{x}) &= \bar{\Theta}^{\bar{r}}_{i_3}(\underline{x})
\end{aligned} \tag{6.29}$$

Once the spatial POD functions $\underline{\underline{\Theta}}^p_i(\underline{x})$, $\underline{\underline{\Theta}}^\alpha_i(\underline{x})$ and $\Theta^{\bar{r}}_i(\underline{x})$ related to the plastic deformation, kinematic hardening and isotropic hardening respectively are determined, the temporal POD functions must be calculated. The following section shows this determination process.

6.3.1.2 Reference points in space: determination of the temporal POD functions

As presented in section 6.2.2, the temporal POD functions of the hyper-reduced approximation are determined by minimizing a reconstruction error defined over some reference points in space $\bar{\underline{x}} = \{\underline{x}_1, \underline{x}_2, ..., \underline{x}_{\bar{n}_{s,g}}\}$. As for the temporal reference points, the number of spatial reference points is given as a fraction of the total integration points in space denoted here $n_{s,g}$, this is:

$$\bar{n}_{s,g} = a_s n_{s,g} \tag{6.30}$$

with the fraction chosen as $a_s = 0.2$. These spatial reference points are chosen as the points where the cumulative plastic deformation is larger, this is:

$$\{\underline{x}_1, \underline{x}_2, ..., \underline{x}_{\bar{n}_{s,g}}\} = \max_{\bar{\underline{x}} \in \Omega_{nl}} \sum_{t \leqslant T} |f_s(\underline{x}, t)| \ ,\ \underline{x} \in \Omega_{nl} \tag{6.31}$$

Once the spatial reference points determined, the temporal functions of the approximation are simply determined by solving the following minimization problems:

$\underline{\bar{x}} \in \Omega_{nl}$,

$$\begin{aligned}
\{\mu_i^p(t)\}_{i=1}^{m_r^p} &= \underset{\{\mu_i^p(t)\}_{i=1}^{m_r^p} \in \mathcal{U}^T}{\arg\min} \left|\left|\left| \sum_{i=1}^{m_r^p} \underline{\underline{\Theta}}_i^p(\underline{\bar{x}})\mu_i^p(t) - \hat{\underline{\underline{\varepsilon}}}^p(\underline{\bar{x}},t) \right|\right|\right|_2^2 \\
\{\mu_i^\alpha(t)\}_{i=1}^{m_r^\alpha} &= \underset{\{\mu_i^\alpha(t)\}_{i=1}^{m_r^\alpha} \in \mathcal{U}^T}{\arg\min} \left|\left|\left| \sum_{i=1}^{m_r^\alpha} \underline{\underline{\Theta}}_i^\alpha(\underline{\bar{x}})\mu_i^\alpha(t) - \hat{\underline{\underline{\alpha}}}(\underline{\bar{x}},t) \right|\right|\right|_2^2 \\
\{\mu_i^{\bar{r}}(t)\}_{i=1}^{m_r^{\bar{r}}} &= \underset{\{\mu_i^{\bar{r}}(t)\}_{i=1}^{m_r^{\bar{r}}} \in \mathcal{U}^T}{\arg\min} \left|\left|\left| \sum_{i=1}^{m_r^{\bar{r}}} \Theta_i^{\bar{r}}(\underline{\bar{x}})\mu_i^{\bar{r}}(t) - \hat{\dot{\bar{r}}}(\underline{\bar{x}},t) \right|\right|\right|_2^2
\end{aligned} \tag{6.32}$$

with the norm:

$$|||\cdot|||_2^2 = \int_{\Omega_r \times I} (\cdot) \overset{\mathsf{D}}{\cdot} (\cdot)\, d\Omega dt \quad , \quad \Omega_r \subset \Omega_{nl}$$

where $\overset{\mathsf{D}}{\cdot}$ correspond to the canonical scalar product between tensors of order D.

The temporal POD functions are obtained following the same resolution procedure as exposed in section 6.2.2. Once the temporal POD functions determined, the approximation of the local stage quantities are given by:

$\forall \underline{x} \in \Omega_{nl},\ \forall t \in I$,

$$\begin{aligned}
\hat{\underline{\underline{\varepsilon}}}^p_{n+1/2}(\underline{x},t) &\approx \sum_{i=1}^{m_r^p} \underline{\underline{\Theta}}_i^p(\underline{x})\mu_i^p(t) \\
\hat{\underline{\underline{\alpha}}}_{n+1/2}(\underline{x},t) &\approx \sum_{i=1}^{m_r^\alpha} \underline{\underline{\Theta}}_i^\alpha(\underline{x})\mu_i^\alpha(t) \\
\hat{\dot{\bar{r}}}_{n+1/2}(\underline{x},t) &\approx \sum_{i=1}^{m_r^{\bar{r}}} \Theta_i^{\bar{r}}(\underline{x})\mu_i^{\bar{r}}(t)
\end{aligned} \tag{6.33}$$

6.3.1.3 Extensibility of the low-rank decomposition defined on the reduced domain Ω_{nl} to the total space domain Ω

It must be remarked that the low-rank approximation presented on the precedent sections is only defined on a reduced domain $\Omega_{nl} \times I$, a domain where the cumulative plastic deformation is not null. In order to extend this decomposition to the total domain $\Omega \times I$ in order to fit correctly for the global stage operation of the LATIN method, the spatial POD functions are filled with zero values at the DOFs corresponding to $\Omega \setminus \Omega_{nl}$, this is, the spatial function are filled such that:

$\forall \underline{x} \in \Omega \setminus \Omega_{nl}$,

$$\{\underline{\underline{\Theta}}_i^p(\underline{x})\}_{i=1}^{m_r^p} = 0, \quad \{\underline{\underline{\Theta}}_i^\alpha(\underline{x})\}_{i=1}^{m_r^\alpha} = 0, \quad \{\Theta_i^{\bar{r}}(\underline{x})\}_{i=1}^{m_r^{\bar{r}}} = 0 \tag{6.34}$$

6.3.1.4 Rank determination of the decomposition

So far we have not discussed how to determine the optimal number of POD modes for the decomposition of all variables in the local stage, in the present section we present an idea for its determination, although different variants can be applied.

The number of POD modes is a very important parameter of the hyper-reduced method, where an optimal number of modes must be determined for each variable, since a low value could produce a bad approximation and a high value could increase the computational cost related to the construction. The procedure considered in this work consists in determining 10 POD modes in an initial computation for each local variable, and then selecting the most relevant ones, such that a given error is verified. This error is defined over additional *indicator points in time* defined on the domain I_{ind}, in where the spatial solution of the local stage variables is completely known. In order to make this an easy and fast algorithm for the optimal mode determination, we consider the number of indicator points in time to be a fraction of the number of reference points, that is:

$$\bar{\bar{n}}_{t,g} = 0.1\ \bar{n}_{t,g} \tag{6.35}$$

This fraction is chosen to be $10[\%]$ of the number of temporal reference points, so this allows a fast algorithm. The error is simply determined as:

$\forall t_j \in I_{ind}$,

$$e^p = 100 \frac{\left\| \sum_{j=1}^{\bar{\bar{n}}_{t,g}} \hat{\underline{\underline{\varepsilon}}}^p(\underline{x}, t_j) - \sum_{i=1}^{m_r^p} \underline{\underline{\Theta}}_i^p(\underline{x}) c_{ij}^p \right\|_\Omega}{\left\| \sum_{j=1}^{\bar{\bar{n}}_{t,g}} \hat{\underline{\underline{\varepsilon}}}^p(\underline{x}, t_j) \right\|_\Omega} \ [\%] \tag{6.36}$$

$$e^\alpha = 100 \frac{\left\| \sum_{j=1}^{\bar{\bar{n}}_{t,g}} \hat{\underline{\underline{\alpha}}}(\underline{x}, t_j) - \sum_{i=1}^{m_r^\alpha} \underline{\underline{\Theta}}_i^\alpha(\underline{x}) c_{ij}^\alpha \right\|_\Omega}{\left\| \sum_{j=1}^{\bar{\bar{n}}_{t,g}} \hat{\underline{\underline{\alpha}}}(\underline{x}, t_j) \right\|_\Omega} \ [\%] \tag{6.37}$$

$$e^{\bar{r}} = 100 \frac{\left\| \sum_{j=1}^{\bar{\bar{n}}_{t,g}} \hat{\bar{r}}(\underline{x}, t_j) - \sum_{i=1}^{m_r^{\bar{r}}} \Theta_i^{\bar{r}}(\underline{x}) c_{ij}^{\bar{r}} \right\|_\Omega}{\left\| \sum_{j=1}^{\bar{\bar{n}}_{t,g}} \hat{\bar{r}}(\underline{x}, t_j) \right\|_\Omega} \ [\%] \tag{6.38}$$

with $\|\cdot\|_\Omega^2 = \int_\Omega (\cdot) \cdot (\cdot)\ d\Omega$.

Due to the orthonormalization of the space POD functions, the different constants c_{ij}^p, c_{ij}^α and $c_{ij}^{\bar{r}}$ are simply calculated by the following projections:

$\forall i_1 \in [1, ..., m_r^p],\ \forall i_2 \in [1, ..., m_r^\alpha],\ \forall i_3 \in [1, ..., m_r^{\bar{r}}],\ \forall t_j \in I_{ind}$,

$$c_{i_1 j}^p = \int_\Omega \underline{\underline{\Theta}}_{i_1}^p(\underline{x}) : \hat{\underline{\underline{\varepsilon}}}^p(\underline{x}, t_j)\ d\Omega \quad ; \quad c_{i_2 j}^\alpha = \int_\Omega \underline{\underline{\Theta}}_{i_2}^\alpha(\underline{x}) : \hat{\underline{\underline{\alpha}}}(\underline{x}, t_j)\ d\Omega \quad ; \quad c_{i_3 j}^{\bar{r}} = \int_\Omega \Theta_{i_3}^{\bar{r}}(\underline{x}) \hat{\bar{r}}(\underline{x}, t_j)\ d\Omega$$

The quantity of POD modes considered for each variable is such that the error estimator for each variable is below 2 $[\%]$, an empirically determined error threshold. An error threshold below 2 $[\%]$ would lead to increasing the computational cost due to the higher amount of POD modes required for its approximation.

6.3.1.5 Error estimator of the approximation

To verify that the low-rank construction is indeed a good approximation of the local stage quantities over the whole space-time domain, an error indicator must be defined. The exact error is impossible to compute, this because to do so, the knowledge of the entire solution over the space-time domain must be known, which is impossible since the low-rank construction only uses the solution over few space-time reference points. This limitation complicates the definition of an exact error indicator and therefore here an estimator of this error is given using only few additional *indicator points in space* defined over Ω_{ind}. Those points are not used for the construction of the decomposition and are only used to calculate an approximated error estimator of the final approximation. The number of these points are selected as a fraction of the total number of reference spatial points, this is:

$$\bar{\bar{n}}_{s,g} = 0.1\ \bar{n}_{s,g} \tag{6.39}$$

In this sense, the error estimator for each local stage variable is defined as follows:

$\forall \underline{x}_j \in \Omega_{ind}$,

$$\begin{aligned}
\epsilon^p &= 100 \frac{\left\| \sum_{j=1}^{\bar{\bar{n}}_{s,g}} \hat{\underline{\underline{\varepsilon}}}^p(\underline{x}_j,t) - \sum_{i=1}^{m_r^p} \underline{\underline{\Theta}}_i^p(\underline{x}_j)\mu_i^p(t) \right\|_2}{\left\| \sum_{j=1}^{\bar{\bar{n}}_{s,g}} \hat{\underline{\underline{\varepsilon}}}^p(\underline{x}_j,t) \right\|_2} \ [\%] \\
\epsilon^\alpha &= 100 \frac{\left\| \sum_{j=1}^{\bar{\bar{n}}_{s,g}} \hat{\underline{\underline{\alpha}}}(\underline{x}_j,t) - \sum_{i=1}^{m_r^\alpha} \underline{\underline{\Theta}}_i^\alpha(\underline{x}_j)\mu_i^\alpha(t) \right\|_2}{\left\| \sum_{j=1}^{\bar{\bar{n}}_{s,g}} \hat{\underline{\underline{\alpha}}}(\underline{x}_j,t) \right\|_2} \ [\%] \\
\epsilon^{\bar{r}} &= 100 \frac{\left\| \sum_{j=1}^{\bar{\bar{n}}_{s,g}} \hat{\bar{r}}(\underline{x}_j,t) - \sum_{i=1}^{m_r^{\bar{r}}} \Theta_i^{\bar{r}}(\underline{x}_j)\mu_i^{\bar{r}}(t) \right\|_2}{\left\| \sum_{j=1}^{\bar{\bar{n}}_{s,g}} \hat{\bar{r}}(\underline{x}_j,t) \right\|_2} \ [\%]
\end{aligned} \tag{6.40}$$

with $\|\cdot\|_2^2 = \int_{\Omega_{ind}\times I} (\cdot) \overset{\mathbb{D}}{:} (\cdot)\, d\Omega dt$.

The above error estimators only serve as indicators to ensure the quality of the hyper-reduced decomposition. A large error estimator would mean that the local variables are not approximated correctly, so the user can move to a classical local stage calculation in order to converge to the desired nonlinear solution.

6.3.2 Global linear stage: Equilibrium and compatibility equations

Once the local stage solution set $\hat{\mathcal{S}}_{n+1/2}$ is computed, the global stage solution set $\mathcal{S}_{n+1}$ must be determined as seen in figure 6.11, and this is done by enriching the already computed solution $\mathcal{S}_n$, this enrichment is done by calculating corrective terms associated to each variable of the solution set. This section shows the principal differences that arise at the global stage when applying the hyper-reduction method at the local stage for the approximation of the constitutive relation quantities, for further details see section 4.4.2.

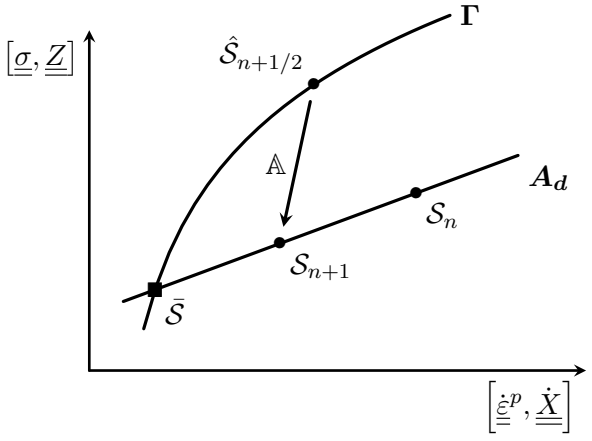


Figure 6.11: Calculation of the global solution of space $\boldsymbol{A_d}$.

The principal variables that must be determined at this step are:

$$\mathcal{S}_{n+1} = \{\underline{u}_{n+1}, \underline{\underline{\varepsilon}}_{n+1}, \underline{\underline{\varepsilon}}^p_{n+1}, \underline{\underline{\sigma}}_{n+1}, \underline{\underline{\beta}}_{n+1}, \underline{\underline{\alpha}}_{n+1}, \bar{r}_{n+1}, \bar{R}_{n+1}\}$$

The solution set $\mathcal{S}_{n+1}$ must also verify the descent search direction as illustrated in figure 6.11, which is given by:

$$\mathbb{A}: \quad \left\{ \begin{bmatrix} \dot{\underline{\underline{\varepsilon}}}^p_{n+1} - \hat{\dot{\underline{\underline{\varepsilon}}}}^p_{n+1/2} \\ -(\dot{\underline{\underline{X}}}_{n+1} - \hat{\dot{\underline{\underline{X}}}}_{n+1/2}) \end{bmatrix} \right. - \mathbb{H}: \left. \begin{bmatrix} \underline{\underline{\sigma}}_{n+1} - \hat{\underline{\underline{\sigma}}}_{n+1/2} \\ \underline{\underline{Z}}_{n+1} - \hat{\underline{\underline{Z}}}_{n+1/2} \end{bmatrix} = 0 \right. \tag{6.41}$$

As presented in chapter 4, the global stage solution at iteration $n+1$ is determined by computing a corrective set $\Delta\mathcal{S}_{n+1}$, such as:

$$\mathcal{S}_{n+1} = \mathcal{S}_n + \Delta\mathcal{S}_{n+1}$$

with:

$$\Delta\mathcal{S}_{n+1} = \{\Delta\underline{u}_{n+1}, \Delta\underline{\underline{\varepsilon}}_{n+1}, \Delta\underline{\underline{\varepsilon}}^p_{n+1}, \Delta\underline{\underline{\sigma}}_{n+1}, \Delta\underline{\underline{\beta}}_{n+1}, \Delta\underline{\underline{\alpha}}_{n+1}, \Delta\bar{r}_{n+1}, \Delta\bar{R}_{n+1}\} \tag{6.42}$$

where the corrective terms are calculated in an inexpensive way by employing the model reduction PGD, which means that the variables at the global stage are approximated as a low-rank decomposition in a separate space-time representation. The low-rank approximation is constructed in two main steps, the enrichment and the preliminary steps. The enrichment step consists in determining the corrective terms of (6.42) as a rank one approximation and the preliminary step consists in actualizing the temporal functions of the PGD decomposition. The corrective terms must be determined such as the constitutive relation errors J^p, J^α and $J^{\bar{r}}$ related to the descend search directions are minimized (see chapter 4), this is:

$$\min J^p = \min \left| \left| \left| \Delta\dot{\underline{\underline{\varepsilon}}}^p_{n+1} - \mathbb{H}_\sigma : \Delta\underline{\underline{\sigma}}_{n+1} + \underline{\underline{\Delta}}_{n+1} \right| \right|\right|^2_{\mathbb{H}^{-1}_\sigma} \tag{6.43}$$

$$\min J^\alpha = \min \left| \left| \left| \Delta\dot{\underline{\underline{\alpha}}}_{n+1} + \mathbb{H}_\beta : \left(C\Delta\underline{\underline{\alpha}}_{n+1} \right) + \underline{\underline{\Delta}}^\alpha_{n+1} \right| \right|\right|^2_{\mathbb{H}^{-1}_\beta} \tag{6.44}$$

$$\min J^{\bar{r}} = \min \left| \left| \left| \Delta\dot{\bar{r}}_{n+1} + H_{\bar{R}} R_\infty \Delta\bar{r}_{n+1} + \Delta^{\bar{r}}_{n+1} \right| \right|\right|^2_{H^{-1}_{\bar{R}}} \tag{6.45}$$

with the residual terms given by:

$$\underline{\underline{\Delta}}_{n+1} = \dot{\underline{\underline{\varepsilon}}}^p_n - \hat{\dot{\underline{\underline{\varepsilon}}}}^p_{n+1/2} \quad ; \quad \underline{\underline{\Delta}}^\alpha_{n+1} = \dot{\underline{\underline{\alpha}}}_n - \hat{\dot{\underline{\underline{\alpha}}}}_{n+1/2} \quad ; \quad \Delta^{\bar{r}}_{n+1} = \dot{\bar{r}}_n - \hat{\dot{\bar{r}}}_{n+1/2}$$

The variables $\dot{\underline{\underline{\varepsilon}}}^p_n$, $\dot{\underline{\underline{\alpha}}}_n$ and $\dot{\bar{r}}_n$ are the global stage solution quantities at LATIN iteration n of the rate of plastic deformation, the rate of kinematic hardening and the rate of isotropic hardening respectively. Therefore they are approximated as a low-rank decomposition by employing the PGD (see chapter 4), this is:

$$\dot{\underline{\underline{\varepsilon}}}^p_n = \sum_{i=1}^m \bar{\underline{\underline{\varepsilon}}}^p_i(\underline{x})\dot{\lambda}_i(t) \quad ; \quad \dot{\underline{\underline{\alpha}}}_n = \sum_{i=1}^m \bar{\underline{\underline{\alpha}}}_i(\underline{x})\dot{\lambda}^\alpha_i(t) \quad ; \quad \dot{\bar{r}}_n = \sum_{i=1}^m \bar{\bar{r}}_i(\underline{x})\dot{\lambda}^{\bar{r}}_i(t)$$

In addition, due to the approximation of the constitutive relation quantities $\hat{\dot{\underline{\underline{\varepsilon}}}}^p_{n+1/2}$, $\hat{\dot{\underline{\underline{\alpha}}}}_{n+1/2}$ and $\hat{\dot{\bar{r}}}_{n+1/2}$ at the local stage by using the hyper-reduction technique, the residual functions associated to each minimization problem are completely given as a low-rank approximation as follows:

$$\underline{\underline{\Delta}}_{n+1} = \sum_{i=1}^m \bar{\underline{\underline{\varepsilon}}}^p_i(\underline{x})\dot{\lambda}_i(t) - \sum_{j=1}^{m^p_r} \underline{\underline{\Theta}}^p_j(\underline{x})\mu^p_j(t) \tag{6.46}$$

$$\underline{\underline{\Delta}}^\alpha_{n+1} = \sum_{i=1}^m \bar{\underline{\underline{\alpha}}}_i(\underline{x})\dot{\lambda}^\alpha_i(t) - \sum_{j=1}^{m^\alpha_r} \underline{\underline{\Theta}}^\alpha_j(\underline{x})\mu^\alpha_j(t) \tag{6.47}$$

$$\Delta^{\bar{r}}_{n+1} = \sum_{i=1}^m \bar{\bar{r}}_i(\underline{x})\dot{\lambda}^{\bar{r}}_i(t) - \sum_{j=1}^{m^{\bar{r}}_r} \Theta^{\bar{r}}_j(\underline{x})\mu^{\bar{r}}_j(t) \tag{6.48}$$

The residual functions of the equations (6.46), (6.47) and (6.48) appear in both the enrichment and preliminary stages, as shown in chapter 4. Therefore, due to the low-rank approximation of the residual functions, the different spatio-temporal integration operations that must be carried out in the global stage, in order to compute a new PGD mode at the enrichment step or improve the decomposition by updating the temporal PGD functions at the preliminary step are drastically optimized. Therefore, the approximation of the local stage by applying the hyper-reduction technique not only decreases the computational cost related to its evaluation, but also decreases the cost of the global stage operations. These computational savings are of great importance when the LATIN-PGD method considers input excitations of long time duration.

6.4 Numerical example

This section presents the numerical results obtained when applying the hyper-reduction technique to the LATIN-PGD method for the elasto-visco-plasticity case and using the TDGM for the temporal resolution. The reference problem corresponds to the same 3D bending beam considered in previous chapters (to show comparable results), which is recalled below in figure 6.12:

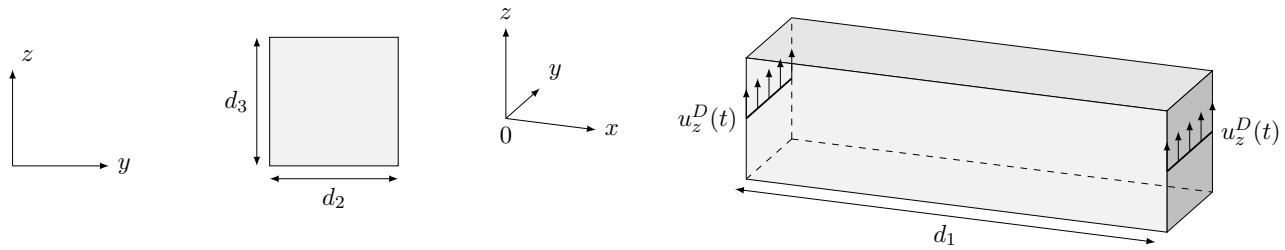


Figure 6.12: Test case considered, along with its dimensions.

The dimension considered are the same as the one presented in chapter 4 for the case of elasto-visco-plasticity, these are:

$$d_1 = 6 \text{ [m]}, \; d_2 = d_3 = 0.1 \text{ [m]}$$

The imposed vertical displacement $u_z^D(t)$ considered for this example corresponds to a signal of relatively large duration and is presented in figure 6.13.

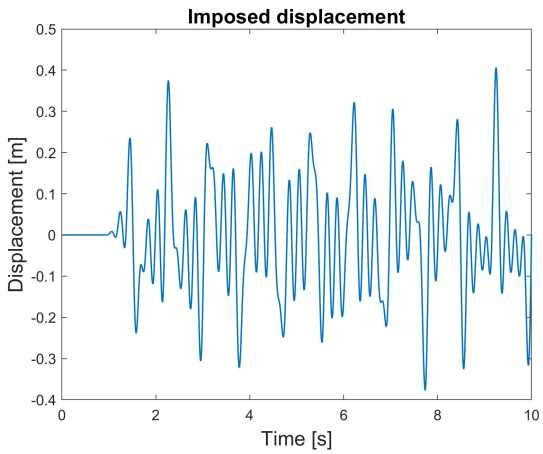


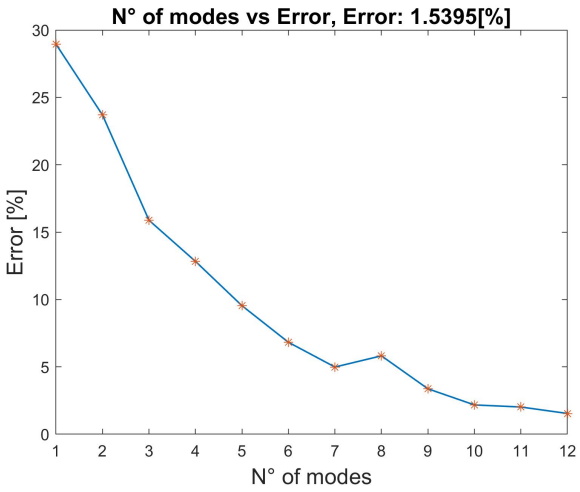
Figure 6.13: Imposed displacement.

For this test the total number of spatial integration points is equal to 6727, however, after the selection of the domain Ω_{nl} (where plasticity is effectively generated) the total number of integration points in space is reduced to 605. On the other hand, the total number of integration points in time considering the above excitation is 4000. In this context, the number of spatio-temporal reference points are selected as presented in section 6.3.1. The information on the number of reference points considered is summarized in the table 6.1.

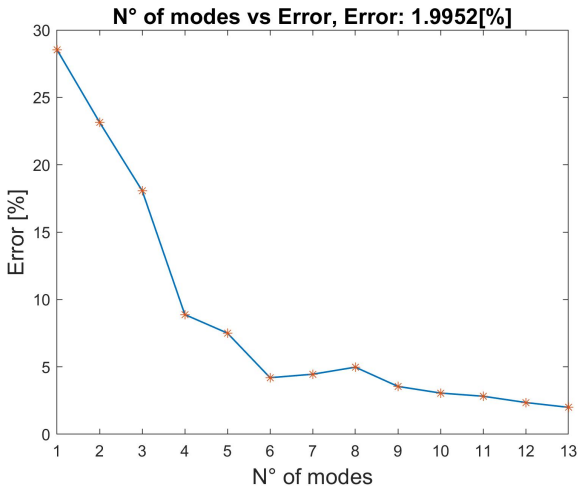
Domain	Integration points	Reference points	Fraction [%]
Space	605	121	20
Time	4000	800	20

Table 6.1: Number of integration points and reference points for the considered numerical test.

Under this conditions, the local stage quantities are well approximated on average by 6 POD modes, where each local stage solution produced an approximated error between 1 $[\%]$ and 2 $[\%]$ for each local variable. The LATIN error of the classical resolution and that obtained with the hyper-reduced method are shown in figure 6.14, these results considered a LATIN error of 2 $[\%]$ as a stopping criterion.



(a) Error using the strategy of chapter 4.



(b) Error using the hyper-reduced strategy.

Figure 6.14: Error comparison between a classic and an hyper-reduced LATIN method when the LATIN error threshold is 2 $[\%]$.

This error is enough for the good approximation of the nonlinear solution when considering an elasto-visco-plastic behavior. Under these considerations the calculation times for the resolution of the reference problem are resumed in table 6.2.

Classic LATIN-PGD	LATIN-PGD + Hyper-reduction	Cost reduction [%]
9.18 minutes	4.04 minutes	56

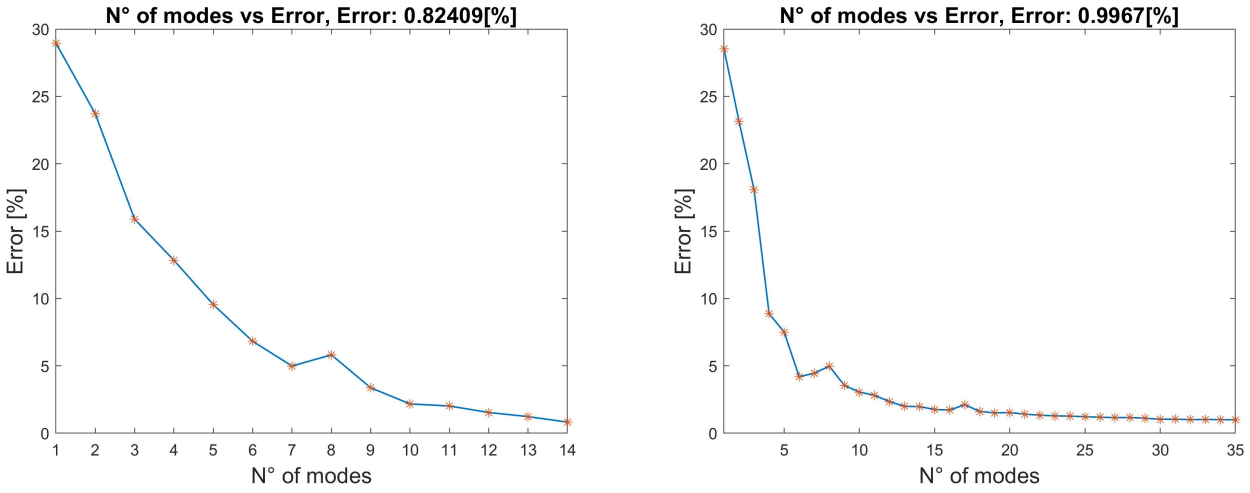
Table 6.2: Comparison of computational time.

From table 6.2, we can observe a 56 $[\%]$ reduction in computational time, a reduction that must be highlighted. However, some numerical problems related to the convergence rate of the hyper-reduced LATIN-PGD method arise when the required error floor is very small.The latter occurs because the low-rank

approximation applied in the local stage captures the main features of the solution, while the local details are less approximated, introducing an associated error. This error affects the resolution of the global stage, which also generates a solution with an associated error, resulting in more iterations being required to converge.The latter occurs because the low-rank approximation applied in the local stage captures the main features of the solution, while the local details are less approximated, introducing an associated error. This error affects the solution of the global stage, which also generates a solution with an associated error, resulting in more iterations being required to converge. This phenomenon results in a higher computational cost due to the additional iterations to be performed, as illustrated in figure 6.15, where a convergence error of 1 $[\%]$ is considered, furthermore, a comparison of the computational cost between the classical and the hyper-reduced LATIN-PGD method for this situation is given in table 6.3.

Classic LATIN-PGD	Hyper-reduced LATIN-PGD
10.18 minutes	11.11 minutes

Table 6.3: Comparison of computational time.



(a) Error using the classic strategy.

(b) Error using the hyper-reduced strategy.

Figure 6.15: LATIN error versus the number of PGD modes between the classical and a hyper-reduced LATIN-PGD method when the LATIN error threshold is 1 $[\%]$.

From the above, it follows naturally that if a low LATIN error is required, the hyper-reduced LATIN-PGD can be used as an accelerator for the first iterations of the solver, while a classical LATIN-PGD method could be used later to reach the required convergence error. In this sense, many different solver strategies can be considered to accelerate resolutions using the hyper-reduced technique presented in this chapter, a technique that opens the door to further developments and improvements.

6.5 Conclusions

Due to the global resolution strategy of the LATIN-PGD method, it requires at each iteration of the method the computation of the constitutive relations in the local stage and then performing costly integration operations in the global stage when computing the space-time PGD functions, this cost increases proportionally with the size of the considered spatio-temporal domain. In order to overcome these limitations, a hyper-reduction technique was introduced in this chapter, that allows to reduce the computational times

for the evaluation of the material constitutive relation and at the same time to approximate it as a low-rank decomposition. This low-rank approximation of the local stage quantities also allows for a more efficient global stage, where the spatial and temporal integrals needed to be performed are computed economically. Of course, the aforementioned advantages of the hyper-reduced strategy are proportional to the size of the spatio-temporal domain considered, motivating its application to problems where the external excitation has a long duration. In this sense, the present hyper-reduction technique is the natural solution when dealing with complex fatigue problems or long seismic inputs.

Despite the advantages of using the hyper-reduced strategy in the local stage, it should be remembered that this technique can only be applied to constitutive relations that can be evaluated locally in the space-time domain. This is not always the case, in fact isotropic damage behavior does not verify this property, therefore a more robust strategy must be developed to consider all types of material behaviors. The domain of artificial intelligence, more specifically deep learning [Goodfellow et al., 2016], could be a possible solution to this limitation, where its universal approximation property can be exploited in order to obtain an approximate constitutive relation that can be evaluated locally in the spatio-temporal domain.

Although the hyper-reduced strategy presented in this chapter has a limited application to certain material behaviors, the author of this thesis strongly believes that the idea of reducing the local stage and approximating it by using a low-rank decomposition is the future of the LATIN-PGD method. This idea helps to solve one of the major disadvantages of the solver, which as mentioned before corresponds to the costly evaluation of the constitutive relation and the integration operations required to compute the PGD approximation in the global stage when the spatial and temporal domains are large, and thus being able to optimize the LATIN-PGD method at the highest level.

Chapter 7

Parallel strategy in time applied to the LATIN-PGD method

In the previous chapters different strategies have been presented to decrease the computational cost related to the time resolution of the LATIN-PGD method. This last chapter goes in the same direction but exploiting a parallel temporal resolution strategy, which arises as a natural consequence of the use of the Time Discontinuous Galerkin method. Numerical results applying this strategy are presented at the end of the chapter, which considers the behavior of an elasto-visco-plastic material.

Contents

7.1 Introduction . . . 164
7.2 Initial elastic problem . . . 166
7.3 Local stage . . . 166
7.4 Global stage: Parallel resolution of the temporal functions . . . 167
7.5 Numerical example . . . 175
7.6 Conclusions . . . 178

7.1 Introduction

Dealing with nonlinear problems defined over a large spatial and temporal domain induces a large consumption of computational resources when the problem is discretized for subsequent solution by a computer. The use of model order reduction techniques allows to reduce the computational cost associated to solving large spatio-temporal problems as seen in chapter 3, but this is not the only way to reduce them, domain decomposition methods are another possibility. Domain decomposition methods also intend to reduce the computational costs by splitting the original (generally space) domain into little subdomains in which the numeric resolution is inexpensive. In addition each subdomain can be solved in parallel by exploiting the architecture of computers by associating a subdomain to each processor, allowing to drastically reduce the time needed to solve the original problem.

Many spatial domain decomposition methods have been developed in recent years, among them we find the Schwarz alternating method [Lions, 1988, Lions, 1990], the additive Schwarz method [Toselli and Widlund, 2006], the Krylov methods [Van der Vorst, 2003], Neumann-Neumann [Bourgat et al., 1988, Mandel, 1993, Le Tallec, 1994], FETI methods [Farhat and Roux, 1992, Farhat et al., 2001], among many others. A spatial domain decomposition strategy was also proposed for the LATIN method (see section 3.4.4), initially presented in [Ladevèze and Lorong, 1991] and further developed in [Champaney et al., 1997, Dureisseix and Ladevèze, 1998, Ladevèze and Dureisseix, 2000, Ladevèze et al., 2001].

The idea of domain decomposition can also be applied to the time domain, but this requires special treatment because time evolution must verify the causality principle, which imposes that every state or solution at time t^* depends on the solution at all times preceding it, i.e., $\forall t < t^*$.

Among the best known time-domain decomposition methods, we find the group of *Multiple Shot Type Methods*. The first implementation of a multiple shooting method is due to Nievergelt in [Nievergelt, 1964], which idea was further improved by [Bellen and Zennaro, 1989] and [Chartier and Philippe, 1993]. The main idea of these methods lies in the decomposition of the temporal domain into macro intervals or temporal subdomains, in where the main objective is to determine the correct initial condition of each subdomain. To do so, the temporal evolution is obtained by solving two problems, a fine problem defined in each temporal subdomain followed by a coarse resolution over the initial conditions of each subdomain. The fine resolution simply consists in solving the problem classically in the time subdomain based on an initial condition that is estimated by the coarse resolution iteratively. The iterative methods seek to determine the correct initial conditions of each time subdomain, once determined, the continuity of the solution in time is ensured.

One of the best known and most widely used method for its robustness and extension to deal with nonlinear problems, which belongs to this kind of methods, is the *Pararéel* method [Lions, 2001]. The main idea consists in the iterative minimization of a temporal jump between a macro and micro solutions. For example, lets consider that a displacement field is obtained, and we denote the semi-discretized coarse displacement solution as $\underline{\boldsymbol{U}}(t)$ and the fine displacement solution as $\underline{\boldsymbol{u}}(t)$, under this considerations the main problem to be solved consists in minimizing the discontinuity between the fine and coarse solutions at the initial conditions of each subdomain, this is:

$$\min \left\| \underline{\boldsymbol{u}}_p^j(T_p) - \underline{\boldsymbol{U}}_p^j \right\|_2 \tag{7.1}$$

with $\|\cdot\|_2$ the euclidean norm, and where j denotes the iterative index that translates how many times the strategy is applied and the index p denotes the temporal subdomain, as shown in figure 7.1. In [Lions, 2001] an implicit Euler method is used for the approximation of the time evolution for the coarse and fine problem ($\underline{\boldsymbol{U}}(t)$ and $\underline{\boldsymbol{u}}(t)$ respectively), that is why the macro solution is illustrated with straight lines in figure 7.1.

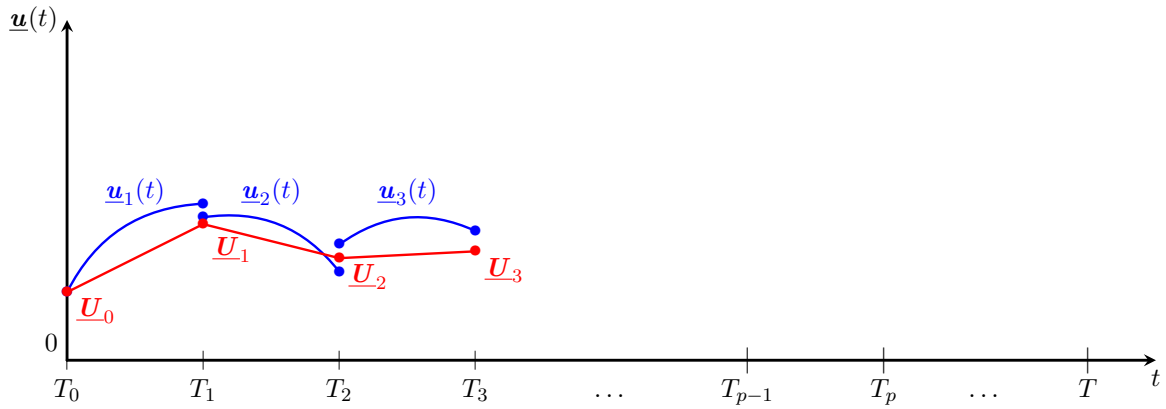


Figure 7.1: Strategy of *Pararéel* method for the temporal resolution.

The *Pararéel* method was developed in such a way that it can be applied to solve problems of scalar time functions or semi-discretized vectors, but despite its popularity and good performance it is difficult to implement in the context of the LATIN-PGD method, due to the need of solving complex minimization problems in time for the calculation of the PGD time functions at the enrichment and preliminary steps of the global stage.

Different works have been developed in the context of domain decomposition applied to the LATIN method to achieve time-parallel resolution. For example, we can cite [Ladevèze and Nouy, 2002, Ladevèze and Nouy, 2003, Nouy, 2003, Néron, 2004, Ladevèze et al., 2007, Passieux et al., 2008, Passieux et al., 2010, Ladevèze et al., 2010a, Ladevèze et al., 2010b], in where a spatio-temporal domain decomposition technique is applied to achieve a fully scalable LATIN solver for the parallel resolution of large spatio-temporal problems. In these works, an homogenization technique in time is applied following a macro-micro temporal resolution strategy (making use of the definition of coarse and fine intervals as in the *Pararéel* method). The macro evolution (defined over the coarse discretization) is assumed to have a smooth evolution such that it can be approximated using dedicated temporal shape functions, while the micro problem (defined on a temporal subdomain) is solved using the Time Discontinuous Galerkin Method (TDGM) of order 0. Both macro and micro problems are solved iteratively during the LATIN resolution, such as they ensure at convergence the continuity between the temporal subdomains.

The aforementioned strategy achieves a temporal parallel resolution in the context of the LATIN-PGD method, however it requires the decomposition of the spatial domain, which increases the complexity of the implementation. In this sense, in the present chapter a different strategy is followed, also based on the Time Discontinuous Galerkin Method but making use of its *incremental* resolution property (see chapter 4), where the idea of macro and micro resolution is reused although from a different point of view. On the one hand, the micro problem is based on an incremental resolution of the functions along the time subdomains considering for each one some given initial conditions, on the other hand, the macro problem consists in the low-cost determination of these initial conditions but unlike the methods exposed above, without assuming any macroevolution. The determination of these values in the macro problem does not require any additional formulation or inversion of operators of any kind, its determination is simply performed incrementally for all temporal subdomains involving only the multiplication of a so-called “transmission operator”. This strategy is easy to implement and its ideas are well adapted to the LATIN-PGD method. However, this strategy is only applicable when constant operators are involved in the descent search directions in the global stage of the LATIN method and when a uniform temporal discretization is used in time (fine mesh). Its application extends both to the initial elastic problem in dynamics and to the solution of the PGD time functions that are determined at each iteration of the method. All the details of this parallel solving strategy are presented in the following sections.

7.2 Initial elastic problem

As discussed in the previous chapters, the LATIN method is initialized with the elastic solution in dynamics $\mathcal{S}_0 \in \boldsymbol{A_d}$, with $\boldsymbol{A_d}$ the admissibility manifold. This solution starts the iterative resolution between the local and global stages, where solutions $\hat{\mathcal{S}}_{n+1/2} \in \boldsymbol{\Gamma}$ and $\mathcal{S}_{n+1} \in \boldsymbol{A_d}$ are calculated respectively (at a given LATIN iteration n), with $\boldsymbol{\Gamma}$ the manifold on which the constitutive relations are verified. This iterative process is recalled below:

$$\mathcal{S}_0 \in \boldsymbol{A_d} \longrightarrow \hat{\mathcal{S}}_{1/2} \in \boldsymbol{\Gamma} ... \longrightarrow \hat{\mathcal{S}}_{n+1/2} \in \boldsymbol{\Gamma} \longrightarrow \mathcal{S}_{n+1} \in \boldsymbol{A_d} ... \longrightarrow \bar{\mathcal{S}} \in \boldsymbol{A_d} \cap \boldsymbol{\Gamma}. \tag{7.2}$$

We recall that, at each iteration of the solver, the distance between the local and global solutions must be minimized, where the method is said to converge when the distance between the two solutions $\hat{\mathcal{S}}_{n+1/2}$ and $\mathcal{S}_{n+1}$ is less than a given threshold.

As shown is chapter 4, the elastic dynamic solution is computed by solving the following equilibrium equation:

$$\underline{\underline{\boldsymbol{M}}}\,\underline{\ddot{\boldsymbol{u}}}_0(t) + \underline{\underline{\boldsymbol{D}}}\,\underline{\dot{\boldsymbol{u}}}_0(t) + \underline{\underline{\boldsymbol{K}}}\,\underline{\boldsymbol{u}}_0(t) = \underline{\boldsymbol{f}}(t)$$

with $\underline{\underline{\boldsymbol{M}}}$, $\underline{\underline{\boldsymbol{D}}}$ and $\underline{\underline{\boldsymbol{K}}}$ the mass, damping and stiffness matrices respectively, and where the vector $\underline{\boldsymbol{u}}_0(t)$ corresponds to the discretized elastic displacement solution at a given time t. As seen in chapter 4, this solution is approximately calculated by using a low-rank decomposition and by computing the elastic solution as a sum of two main terms, a quasi-static and a dynamic term:

$$\underline{u}_0(\underline{x},t) = \underbrace{\underline{u}_{0,q}(\underline{x},t)}_{\text{quasi-static term}} + \underbrace{\underline{u}_{0,d}(\underline{x},t)}_{\text{dynamic term}}$$

Where the quasi-static term is aproached by using the PGD and the dynamic term by using the modal basis of the structure:

$$\underline{u}_{0,q}(t) \approx \sum_{i=1}^{m_q} (\underline{w}_q)_i(\underline{x})(\lambda_q)_i(t) \qquad \text{and} \qquad \underline{u}_{0,d}(t) \approx \sum_{i=1}^{m_b} \underline{\phi}_i(\underline{x})\upsilon_i(t)$$

where $\underline{\phi}_i(\underline{x})$ and $\upsilon_i(t)$ correspond to the spatial and temporal modal basis functions, while $(\underline{w}_q)_i(\underline{x})$ and $(\lambda_q)_i(t)$ correspond to the spatial and temporal functions of the PGD applied to the quasi-static term, being m_q and m_b the rank of the quasi-static and dynamic term approximation. Due to the PGD and the use of the modal basis approximation, the temporal evolution of the elastic solution simply consists in the calculation of scalar temporal functions (once the spatial ones are determined). The temporal functions associated to the quasi-static and dynamic terms can be computed in parallel by using the same methodology that will be introduced in section 7.4. In this regard, the presentation of the elastic resolution is omitted to avoid re-explanation of the strategy, however, the application of the parallel strategy to the elastic problem is straightforward and does not impose any additional difficulty.

7.3 Local stage

The local stage of the LATIN-PGD method remains unchanged, the degree of parallelization of this stage depends on the formulation of the constitutive relation itself. One recalls here that the nonlinear part of the constitutive relations must be evaluated for each Gauss point in time and space. Some constitutive relations can be computed easily in a parallel way, this is the case for instance of elasto-visco-plasticity (see section 2.3.2) because of its explicitly calculation in space and time; indeed, each point of the space-time domain is calculated in an independent way from one another. On the other hand the constitutive

relation associated to isotropic damage (see section 2.3.1) can be calculated in parallel for each spatial point but can not be calculated in parallel for the temporal domain due to the need of the resolution of an ordinary differential equation in time. Many different constitutive relations are formulated in different ways, so therefore the degree of parallelization must be evaluated for each particular case.

7.4 Global stage: Parallel resolution of the temporal functions

After the evaluation of the constitutive relation at the local stage where the solution $\hat{\mathcal{S}}_{n+1/2} \in \mathbf{\Gamma}$ is determined, a linear problem is solved at the global stage where the solution $\mathcal{S}_{n+1} \in \boldsymbol{A_d}$ must be computed. As presented in chapter 4, the global stage quantities are approximated by employing the model reduction technique PGD, the construction of this low-rank approximation is done in two steps, the *enrichment* and the *preliminary* steps. The enrichment step consists in enriching or adding a new PGD mode (space-time couple functions) to the already low-rank decomposition and the preliminary step consists in improving this low-rank representation by actualizing the temporal PGD functions (while keeping the spatial functions fixed).

In the present section, a parallel strategy for the computation of the temporal PGD functions of the global stage is presented. The parallel methodology is adapted to both enrichment and preliminary steps. Herein we will consider only the step where the temporal functions are calculated without detailing how the equations are obtained. For more details about the developments of the different equations, please refer to chapter 4.

Before showing the details of the parallel strategy applied to the enrichment and the preliminary steps, it is necessary to introduce the key notation that will be used throughout the following sections, as well as the general idea for achieving parallel computation, these are presented below.

1. **Macro-micro temporal discretization**:

   Lets consider that the total interval of time $I = [0, T]$ is divided into N macro temporal subdomains, this is:

   $$I_p = [T_{p-1}, T_p] \ , \ \forall p \in [1, 2, ..., N] \tag{7.3}$$

   with $T_0 = 0$ and $T_N = T$. Also lets define the maximum number of temporal finite elements inside I_p as $N_T^{(p)}$, as illustrated in figure 7.2, where the index k_p denotes a temporal FEM element on the macro interval "p".

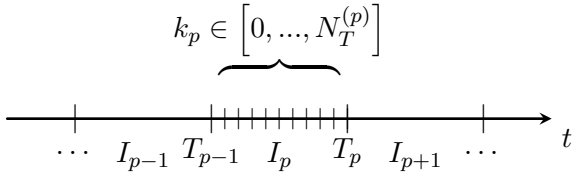


Figure 7.2: Macro-micro temporal discretization.

   In this chapter, the **macro subdomains are considered to be uniform as well as the micro discretization**, in this sense, it must be noted that $N-1$ temporal subdomains have the same size, furthermore, this implies that the number of finite elements within those uniform subdomains are equal, in other words:

   $\forall p \in [1, 2, ..., N-2]$,

   $$N_T^{(p+1)} = N_T^{(p)} \tag{7.4}$$

The only subdomain with a size eventually different from the others is the last one $p = N$, which is illustrated in figure 7.3.

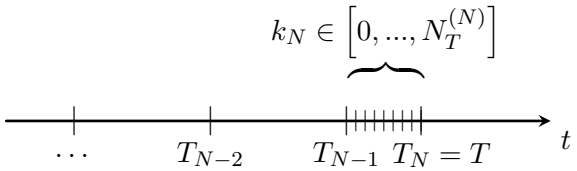


Figure 7.3: Last temporal subdomain.

Considering uniform macro and micro discretizations brings a great advantage, since many operations of the parallel strategy that will be introduced are simplified and operators are reused, decreasing by consequence the overall computational cost of the technique.

2. **Micro and macro problems**:

   Once the time domain $I = [0, T]$ is divided into a few subdomains, the parallel strategy of resolution simply consists in solving 3 main problems, **two micro** and **one macro** problems. For the presentation of these problems we define the following notation for the temporal function:

$$\lambda_{\ell,p}(t) \tag{7.5}$$

   where the index ℓ represents the number of micro problems to be solved and p denotes the temporal subdomain. The main idea of these three problems are explained below.

   (a) **First micro resolution** ($\ell = 1$):

   The first stage of the parallel strategy is to set all initial conditions in each time subdomain equal to zero and then solve the functions using the TDGM in each subdomain independently. This process is illustrated in figure 7.4.

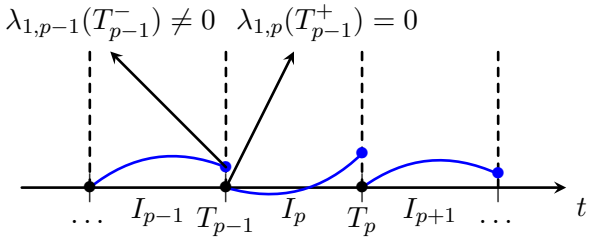


Figure 7.4: First micro problem, all the initial conditions of the temporal subdomains set equals to 0.

   At the end of this step, the temporal function obtained over the entire time domain is completely discontinuous as illustrated in figure 7.4, in order to re-impose continuity a macro resolution is performed, which is presented below.

   (b) **Macro resolution**:

   Once the first fine problem is solved, the time functions suffer from high discontinuities, to solve this, the final values in each subdomain are updated following a serial resolution over the macro discretization as shown in figure 7.5.

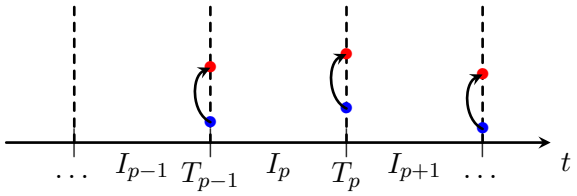


Figure 7.5: Macro problem defined on each temporal subdomain.

In the following sections it will be shown how the end values in each subdomain are actualized by solving this macro problem. In addition, it will be shown that **by using the TDGM the correct initial conditions on each subdomain are obtained by solving this problem only once**.

(c) **Second micro resolution** ($\ell = 2$):

Once the macro problem is solved, the correct values of the initial conditions in each time subdomain are obtained, so the last step is to solve again a micro problem but this time considering the updated initial conditions. At the end of this step the global continuity of the time function is obtained as illustrated in figure 7.6.

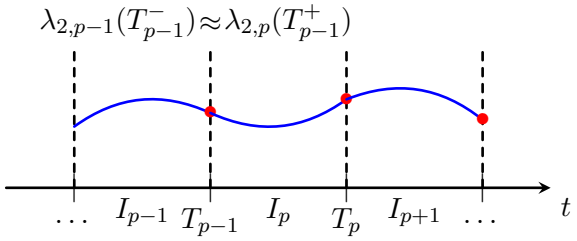


Figure 7.6: Second micro problem, recovery of the continuity.

From the above, it should be noted that the “micro” problems can be solved completely independently since all the initial conditions are known, so that their calculation can be performed in parallel. On the other hand, the “macro” problem must be solved serially and incrementally.

In the next sections, the parallel strategy applied for the resolution of the temporal PGD functions associated to the enrichment and preliminary steps is presented. Only the details related to the temporal resolution will be exposed, omitting therefore some details such as the spatial problem of the enrichment step or how the discretized equations are obtained in time, for more details on these points see chapter 4.

7.4.1 Enrichment step: temporal function determination

The enrichment step consists in the computation of a spatio-temporal pair function usually denoted as PGD mode. For its computation a spatial and a temporal problem must be solved iteratively, however in this section only the temporal problem is studied. The temporal PGD function in the enrichment step is determined by minimizing the constitutive relation error, this error is defined according to the behavior of the material considered, for example for isotropic damage this error is given by:

$$\{\lambda_{m+1}\} = \underset{\lambda_{m+1} \in \mathcal{U}^T}{\arg\min} \left|\left|\left| \bar{\underline{\underline{\sigma}}}_{m+1} \lambda_{m+1} - \mathbb{H}_\varepsilon : \bar{\underline{\underline{\varepsilon}}}_{m+1} \lambda_{m+1} + \underline{\underline{\Delta}}_{n+1} \right|\right|\right|^2_{\mathbb{H}_\varepsilon^{-1}} \tag{7.6}$$

while for elasto-visco-plasticity by:

$$\begin{aligned}
\{\lambda_{m+1}\} &= \underset{\lambda_{m+1}\in\mathcal{U}^T}{\text{arg min}} \left|\left|\left| \dot{\bar{\underline{\underline{\varepsilon}}}}^p_{m+1}\dot{\lambda}_{m+1} - \mathbb{H}_\sigma : \bar{\underline{\underline{\sigma}}}_{m+1}\lambda_{m+1} + \underline{\underline{\Delta}}_{n+1} \right|\right|\right|^2_{\mathbb{H}_\sigma^{-1}} \\
\{\lambda^\alpha_{m+1}\} &= \underset{\lambda^\alpha_{m+1}\in\mathcal{U}^T}{\text{arg min}} \left|\left|\left| \bar{\underline{\underline{\alpha}}}_{m+1}\dot{\lambda}^\alpha_{m+1} + \mathbb{H}_\beta : C\bar{\underline{\underline{\alpha}}}_{m+1}\lambda^\alpha_{m+1} + \underline{\underline{\Delta}}^\alpha_{n+1} \right|\right|\right|^2_{\mathbb{H}_\beta^{-1}} \\
\{\lambda^{\bar{r}}_{m+1}\} &= \underset{\lambda^{\bar{r}}_{m+1}\in\mathcal{U}^T}{\text{arg min}} \left|\left|\left| \bar{\bar{r}}_{m+1}\dot{\lambda}^{\bar{r}}_{m+1} + H_{\bar{R}}R_\infty\bar{\bar{r}}_{m+1}\lambda^{\bar{r}}_{m+1} + \Delta^{\bar{r}}_{n+1} \right|\right|\right|^2_{H_{\bar{R}}^{-1}}
\end{aligned} \tag{7.7}$$

After discretization of the minimization problems of equations (7.6) or (7.7) by using the TDGM as presented in chapter 4, the following equation can be obtained without loss of generality:

$$\left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\mathcal{L}}}\right)\underline{\boldsymbol{\lambda}}^{[k]} = \underline{\underline{\mathcal{R}}}\underline{\boldsymbol{\lambda}}^{[k-1]} + \underline{\boldsymbol{f}}^{[k]} \tag{7.8}$$

with the matrix $\underline{\underline{\boldsymbol{Q}}}$ corresponding to the finite element discretization using TDGM approach, while $\underline{\underline{\mathcal{L}}}$ and $\underline{\underline{\mathcal{R}}}$ correspond to the matrices that help to ensure the continuity in a weak sense from one element to another. Lets remark that the elemental index "k" used in chapter 4 is dropped off, this is, $\underline{\underline{\boldsymbol{Q}}} = \underline{\underline{\boldsymbol{Q}}}^{[k]}$, $\underline{\underline{\mathcal{L}}} = \underline{\underline{\mathcal{L}}}^{[k]}$ and $\underline{\underline{\mathcal{R}}} = \underline{\underline{\mathcal{R}}}^{[k]}$, since these matrices have entry values which are constants along the whole temporal domain due to the choice of a uniform micro discretization and the choice of constants operators $\mathbb{H}_\varepsilon$, $\mathbb{H}_\sigma$, $\mathbb{H}_\beta$ and $H_{\bar{R}}$.

Equation (7.8) gives the solution at the element k from the knowledge of the solution at $k-1$, however this expression is not very useful if we consider to develop a parallel strategy. In this sense, in order to obtain a more useful expression, we seek the solution at element k from an initial value in the temporal domain, i.e., element $k=0$. To do so, lets calculate the nodal values of the temporal function at a given element k recursively, in this way the solution of the first element is given by:

$$\underline{\boldsymbol{\lambda}}^{[1]} = \left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\mathcal{L}}}\right)^{-1}\underline{\underline{\mathcal{R}}}\underline{\boldsymbol{\lambda}}^{[0]} + \left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\mathcal{L}}}\right)^{-1}\underline{\boldsymbol{f}}^{[1]}$$

In the same way, the solution at the second element ($k=2$) is given by:

$$\underline{\boldsymbol{\lambda}}^{[2]} = \left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\mathcal{L}}}\right)^{-1}\underline{\underline{\mathcal{R}}}\underline{\boldsymbol{\lambda}}^{[1]} + \left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\mathcal{L}}}\right)^{-1}\underline{\boldsymbol{f}}^{[2]}$$

By introducing the solution of element $k=1$ into $k=2$ we obtain:

$$\underline{\boldsymbol{\lambda}}^{[2]} = \left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\mathcal{L}}}\right)^{-1}\underline{\underline{\mathcal{R}}}\left(\left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\mathcal{L}}}\right)^{-1}\underline{\underline{\mathcal{R}}}\underline{\boldsymbol{\lambda}}^{[0]} + \left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\mathcal{L}}}\right)^{-1}\underline{\boldsymbol{f}}^{[1]}\right) + \left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\mathcal{L}}}\right)^{-1}\underline{\boldsymbol{f}}^{[2]}$$

which can be expressed as:

$$\underline{\boldsymbol{\lambda}}^{[2]} = \left[\left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\mathcal{L}}}\right)^{-1}\underline{\underline{\mathcal{R}}}\right]^2\underline{\boldsymbol{\lambda}}^{[0]} + \left(\left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\mathcal{L}}}\right)^{-1}\underline{\underline{\mathcal{R}}}\right)\left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\mathcal{L}}}\right)^{-1}\underline{\boldsymbol{f}}^{[1]} + \left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\mathcal{L}}}\right)^{-1}\underline{\boldsymbol{f}}^{[2]}$$

The above steps can be recursively generalized to an arbitrary element k as follows:

$$\begin{aligned}
\underline{\boldsymbol{\lambda}}^{[1]} &= \left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\mathcal{L}}}\right)^{-1}\underline{\underline{\mathcal{R}}}\underline{\boldsymbol{\lambda}}^{[0]} + \left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\mathcal{L}}}\right)^{-1}\underline{\boldsymbol{f}}^{[1]} \\
\underline{\boldsymbol{\lambda}}^{[2]} &= \left[\left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\mathcal{L}}}\right)^{-1}\underline{\underline{\mathcal{R}}}\right]^2\underline{\boldsymbol{\lambda}}^{[0]} + \left(\left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\mathcal{L}}}\right)^{-1}\underline{\underline{\mathcal{R}}}\right)\left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\mathcal{L}}}\right)^{-1}\underline{\boldsymbol{f}}^{[1]} + \left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\mathcal{L}}}\right)^{-1}\underline{\boldsymbol{f}}^{[2]} \\
&\vdots \\
\underline{\boldsymbol{\lambda}}^{[k]} &= \left[\left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\mathcal{L}}}\right)^{-1}\underline{\underline{\mathcal{R}}}\right]^k\underline{\boldsymbol{\lambda}}^{[0]} + \sum_{i=1}^{k}\left[\left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\mathcal{L}}}\right)^{-1}\underline{\underline{\mathcal{R}}}\right]^{k-i}\left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\mathcal{L}}}\right)^{-1}\underline{\boldsymbol{f}}^{[i]}
\end{aligned} \tag{7.9}$$

with $\left(\left(\underline{\underline{\boldsymbol{Q}}}+\underline{\underline{\boldsymbol{\mathcal{L}}}}\right)^{-1}\underline{\underline{\boldsymbol{\mathcal{R}}}}\right)^{0}$ equals the identity matrix $\underline{\underline{\boldsymbol{I}}}\in\mathbb{R}^4\otimes\mathbb{R}^4$ (with 4 the elementary number of nodal values in time when using cubic Hermite shape functions).

Therefore, from equation (7.9) the solution of the discretized temporal PGD function at temporal element k is given by:

$$\underline{\boldsymbol{\lambda}}^{[k]}=\left[\left(\underline{\underline{\boldsymbol{Q}}}+\underline{\underline{\boldsymbol{\mathcal{L}}}}\right)^{-1}\underline{\underline{\boldsymbol{\mathcal{R}}}}\right]^{k}\underline{\boldsymbol{\lambda}}^{[0]}+\sum_{j=1}^{k}\left[\left(\underline{\underline{\boldsymbol{Q}}}+\underline{\underline{\boldsymbol{\mathcal{L}}}}\right)^{-1}\underline{\underline{\boldsymbol{\mathcal{R}}}}\right]^{k-j}\left(\underline{\underline{\boldsymbol{Q}}}+\underline{\underline{\boldsymbol{\mathcal{L}}}}\right)^{-1}\underline{\boldsymbol{f}}^{[j]} \tag{7.10}$$

Equation (7.10) allows to compute the solution at any given time element k as the sum of two terms, one associated to the initial condition and other related to a particular solution:

$$\underline{\boldsymbol{\lambda}}^{[k]}=\underbrace{\left[\left(\underline{\underline{\boldsymbol{Q}}}+\underline{\underline{\boldsymbol{\mathcal{L}}}}\right)^{-1}\underline{\underline{\boldsymbol{\mathcal{R}}}}\right]^{k}\underline{\boldsymbol{\lambda}}^{[0]}}_{\text{Initial condition term}}+\underbrace{\sum_{j=1}^{k}\left[\left(\underline{\underline{\boldsymbol{Q}}}+\underline{\underline{\boldsymbol{\mathcal{L}}}}\right)^{-1}\underline{\underline{\boldsymbol{\mathcal{R}}}}\right]^{k-j}\left(\underline{\underline{\boldsymbol{Q}}}+\underline{\underline{\boldsymbol{\mathcal{L}}}}\right)^{-1}\underline{\boldsymbol{f}}^{[j]}}_{\text{Particular solution}}$$

The above expression intrinsically hides the parallel methodology that will be presented in the following sections and is one of the heart equations of the entire chapter.

7.4.1.1 Parallel resolution

The parallel strategy simply begins by considering the splitting of the original time domain $I=[0,T]$ into macro subdomains as presented at the beginning of the section and by applying on each of them the same idea of equation (7.10), this is:

$\forall\ell\in[1,2],\ \forall k_p\in\left[0,...,N_T^{(p)}\right],\ \forall p\in[1,2,...,N]\,,$

$$\underline{\boldsymbol{\lambda}}_{\ell,p}^{[k_p]}=\left[\left(\underline{\underline{\boldsymbol{Q}}}+\underline{\underline{\boldsymbol{\mathcal{L}}}}\right)^{-1}\underline{\underline{\boldsymbol{\mathcal{R}}}}\right]^{k_p}\underline{\boldsymbol{\lambda}}_{\ell,p}^{[0]}+\sum_{j=1}^{k_p}\left[\left(\underline{\underline{\boldsymbol{Q}}}+\underline{\underline{\boldsymbol{\mathcal{L}}}}\right)^{-1}\underline{\underline{\boldsymbol{\mathcal{R}}}}\right]^{k-j}\left(\underline{\underline{\boldsymbol{Q}}}+\underline{\underline{\boldsymbol{\mathcal{L}}}}\right)^{-1}\underline{\boldsymbol{f}}_{p}^{[j]} \tag{7.11}$$

where $\underline{\boldsymbol{f}}_p$ corresponds to the external excitation vector defined within I_p. With the use of equation (7.11) the parallel strategy simply stands as follows:

1. **First micro problem** ($\ell=1$):

   Solve the temporal problem for all the subdomains by imposing an initial condition equal to 0 on all of them, this is:

   $\forall p\in[1,2,...,N]\,,$

   $$\underline{\boldsymbol{\lambda}}_{1,p}^{[0]}=0$$

   this means that the solution $\underline{\boldsymbol{\lambda}}_{1,p}$ is given by:

   $\forall k_p\in\left[0,...,N_T^{(p)}\right],\ \forall p\in[1,2,...,N]\,,$

   $$\underline{\boldsymbol{\lambda}}_{1,p}^{[k_p]}=\sum_{j=1}^{k_p}\left[\left(\underline{\underline{\boldsymbol{Q}}}+\underline{\underline{\boldsymbol{\mathcal{L}}}}\right)^{-1}\underline{\underline{\boldsymbol{\mathcal{R}}}}\right]^{k_p-j}\left(\underline{\underline{\boldsymbol{Q}}}+\underline{\underline{\boldsymbol{\mathcal{L}}}}\right)^{-1}\underline{\boldsymbol{f}}_{p}^{[j]} \tag{7.12}$$

   Since the initial conditions of all considered temporal subdomains are set equal to zero, the time function can be calculated following the equation (7.12) in parallel form, where the time function is determined independently in each time subdomain and thus its resolution can be assigned to different computer processors.

2. **Macro problem**:

Once the first micro problem ($\ell = 1$) is solved, the initial conditions of each subdomain is actualized for the second micro problem ($\ell = 2$) by solving a macro problem. This macro problem is obtained by considering equation (7.11) and by imposing to obtain the solution at the second micro problem ($\ell = 2$) and at the last temporal element on each subdomain p ($k_p = N_T^{(p)}$), this is:

$\forall p \in [1, 2, ..., N-1]$,

$$\underline{\boldsymbol{\lambda}}_{2,p}^{[N_T^{(p)}]} = \left[\left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\boldsymbol{\mathcal{L}}}}\right)^{-1} \underline{\underline{\boldsymbol{\mathcal{R}}}}\right]^{N_T^{(p)}} \underline{\boldsymbol{\lambda}}_{2,p}^{[0]} + \sum_{j=1}^{N_T^{(p)}} \left[\left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\boldsymbol{\mathcal{L}}}}\right)^{-1} \underline{\underline{\boldsymbol{\mathcal{R}}}}\right]^{k-j} \left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\boldsymbol{\mathcal{L}}}}\right)^{-1} \underline{\boldsymbol{f}}_p^{[j]} \tag{7.13}$$

By remembering that due to the use of the TDGM we have $\underline{\boldsymbol{\lambda}}_{2,p+1}^{[0]} = \underline{\boldsymbol{\lambda}}_{2,p}^{[N_T^{(p)}]}$ (the solution on the last element of subdomain p is the initial condition of subdomain $p+1$) and by recognizing the result of (7.12) in the right side term of (7.13), we can write:

$\forall p \in [1, 2, ..., N-1]$,

$$\underline{\boldsymbol{\lambda}}_{2,p+1}^{[0]} = \underline{\boldsymbol{\lambda}}_{2,p}^{[N_T^{(p)}]} = \underbrace{\left[\left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\boldsymbol{\mathcal{L}}}}\right)^{-1} \underline{\underline{\boldsymbol{\mathcal{R}}}}\right]^{N_T^{(p)}} \underline{\boldsymbol{\lambda}}_{2,p}^{[0]}}_{\underline{\underline{\boldsymbol{P}}}\,\underline{\boldsymbol{\lambda}}_{2,p}^{[0]}} + \underbrace{\sum_{j=1}^{N_T^{(p)}} \left[\left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\boldsymbol{\mathcal{L}}}}\right)^{-1} \underline{\underline{\boldsymbol{\mathcal{R}}}}\right]^{k-j} \left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\boldsymbol{\mathcal{L}}}}\right)^{-1} \underline{\boldsymbol{f}}_p^{[j]}}_{\underline{\boldsymbol{\lambda}}_{1,p}^{[N_T^{(p)}]}} \tag{7.14}$$

where the matrix $\underline{\underline{\boldsymbol{P}}}$ is denoted the "transmission operator", which is a constant matrix given by:

$$\underline{\underline{\boldsymbol{P}}} = \left[\left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\boldsymbol{\mathcal{L}}}}\right)^{-1} \underline{\underline{\boldsymbol{\mathcal{R}}}}\right]^{N_T^{(1)}}, \ \underline{\underline{\boldsymbol{P}}} \in \mathbb{R}^4 \otimes \mathbb{R}^4 \tag{7.15}$$

where in the above expression we recall that $N_T^{(1)} = N_T^{(p)}$, $\forall p \in [1, 2, ..., N-1]$, since the micro and macro discretization are chosen to be uniform.

With all the above considerations, the macro problem can by simply stated as follows:

$\forall p \in [1, 2, ..., N-1]$,

$$\underline{\boldsymbol{\lambda}}_{2,p+1}^{[0]} = \underline{\underline{\boldsymbol{P}}}\,\underline{\boldsymbol{\lambda}}_{2,p}^{[0]} + \underline{\boldsymbol{\lambda}}_{1,p}^{[N_T^{(p)}]} \quad \text{with} \quad \underline{\boldsymbol{\lambda}}_{2,p=1}^{[0]} = 0 \quad \text{(Initial condition at t=0)} \tag{7.16}$$

Equation (7.16) allows determining the correct initial condition for each time subdomain by incrementally solving one subdomain after another, where in each subdomain the constant transmission operator is used together with the final solution of the first micro problem ($\ell = 1$). Figure 7.7 illustrates this solving process.

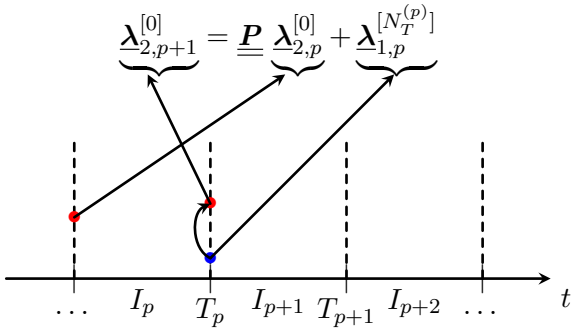


Figure 7.7: Solving the macro problem: Updating the initial conditions for all subdomains of the second micro problem (red dots) from the final value in each subdomain of the first micro solution (blue dots).

Remark: The computation of the transmission operator requires the evaluation of a matrix raised to the power of $N_T^{(1)}$, which is the total number of finite elements over time in each uniform time subdomain. Note that the choice of uniform subdomains implies a constant matrix transmission operator $\underline{\underline{\boldsymbol{P}}}$ that can be reused throughout the resolution of the macro problem. It should also be noted that the solution of the macro problem is performed incrementally for each time subdomain (serially), so the macro problem cannot be solved in parallel, however its solution is inexpensive due to the constant transmission operator used, which reduces the solution of the macro problem to the realization of inexpensive matrix-vector multiplication operations for the determination of the new initial conditions in each subdomain. Finally, it should be noted that at the end of the solution of the macro problem, the new initial conditions determined in each time subdomain are actually the initial conditions of the converging time function, this is because their determination verifies exactly the equation of the time solution 7.11. In this sense, after the solution of the macro problem, only an additional solution of the micro problem is required to obtain the desired time function (which verifies the continuity of the function in the TDGM sense).

3. **Second micro problem** ($\ell = 2$):

   Once the correct initial conditions for all the subdomains are determined at the macro problem, the solution of the second micro problem $\underline{\boldsymbol{\lambda}}_{2,p}$ is simply determined over each subdomain as:

   $\forall k_p \in \left[0, ..., N_T^{(p)}\right],\ \forall p \in [1, 2, ..., N]\,,$

$$\underline{\boldsymbol{\lambda}}_{2,p}^{[k_p]} = \left[\left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\boldsymbol{\mathcal{L}}}}\right)^{-1} \underline{\underline{\boldsymbol{\mathcal{R}}}}\right]^{k_p} \underline{\boldsymbol{\lambda}}_{2,p}^{[0]} + \sum_{j=1}^{k_p} \left[\left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\boldsymbol{\mathcal{L}}}}\right)^{-1} \underline{\underline{\boldsymbol{\mathcal{R}}}}\right]^{k-j} \left(\underline{\underline{\boldsymbol{Q}}} + \underline{\underline{\boldsymbol{\mathcal{L}}}}\right)^{-1} \underline{\boldsymbol{f}}_p^{[j]} \tag{7.17}$$

   Since the determination of the initial condition at the macro problem correspond to the exact initial condition as if a classic TDGM is used, the final temporal solution $\underline{\boldsymbol{\lambda}}$ simply corresponds to the union of all the computed solutions at this step, finally obtaining:

$$\underline{\boldsymbol{\lambda}} = \bigcup_{p=1}^{N} \underline{\boldsymbol{\lambda}}_{2,p} \tag{7.18}$$

7.4.2 Preliminary step: Temporal functions actualization

As stated in precedents chapters, an improvement in the convergence of the LATIN method is achieved if the update of the temporal PGD functions is done at the preliminary step. The update of the temporal functions is done by simply considering that the temporal PGD functions of the global variables can be corrected as:

$\forall i \in [1, ..., m+1]$,

$$\lambda_i \leftarrow \lambda_i + \Delta\lambda_i \tag{7.19}$$

The expression (7.19) is true for the case of the admissibility variables (displacement, stress, total deformation, etc), for particular cases as internal variables, it is chosen to simply correct the last PGD mode in order to avoid bad conditioning (see chapter 4):

$$\lambda_{m+1}^{\alpha} \leftarrow \lambda_{m+1}^{\alpha} + \Delta\lambda_{m+1}^{\alpha} \tag{7.20}$$

$$\lambda_{m+1}^{\bar{r}} \leftarrow \lambda_{m+1}^{\bar{r}} + \Delta\lambda_{m+1}^{\bar{r}} \tag{7.21}$$

By introducing those expression into the different descent search directions (which are function of the material considered) we simply obtain the following minimization problems for the case of isotropic damage:

$\forall i \in [1, ..., m+1]$,

$$\{\Delta\lambda_i\} = \underset{\{\Delta\lambda_i\}_{i=1}^m \in \mathcal{U}^T}{\arg\min} \left|\left|\left| \sum_{i=1}^{m+1} \bar{\underline{\underline{\sigma}}}_i \Delta\lambda_i - \mathbb{H}_\varepsilon : \left(\sum_{i=1}^{m+1} \bar{\underline{\underline{\varepsilon}}}_i \Delta\lambda_i \right) + \underline{\underline{\Delta}}_{n+1} \right|\right|\right|_{\mathbb{H}_\varepsilon^{-1}}^2 \tag{7.22}$$

while for elasto-visco-plasticity we have:

$\forall i \in [1, ..., m+1]$,

$$\{\Delta\lambda_i\} = \underset{\{\Delta\lambda_i\}_{i=1}^m \in \mathcal{U}^T}{\arg\min} \left|\left|\left| \sum_{i=1}^{m+1} \bar{\underline{\underline{\varepsilon}}}_i^p \Delta\dot{\lambda}_i - \mathbb{H}_\sigma : \sum_{i=1}^{m+1} \bar{\underline{\underline{\sigma}}}_i \Delta\lambda_i + \underline{\underline{\Delta}}_{n+1} \right|\right|\right|_{\mathbb{H}_\sigma^{-1}}^2 \tag{7.23}$$

with the minimization problem related to the internal variables given by:

$$\{\Delta\lambda_{m+1}^\alpha\} = \underset{\Delta\lambda_{m+1}^\alpha \in \mathcal{U}^T}{\arg\min} \left|\left|\left| \bar{\underline{\underline{\alpha}}}_{m+1} \Delta\dot{\lambda}_{m+1}^\alpha - \mathbb{H}_\beta : C \bar{\underline{\underline{\alpha}}}_{m+1} \Delta\lambda_{m+1}^\alpha + \underline{\underline{\Delta}}_{n+1}^\alpha \right|\right|\right|_{\mathbb{H}_\beta^{-1}}^2 \tag{7.24}$$

$$\{\Delta\lambda_{m+1}^{\bar{r}}\} = \underset{\Delta\lambda_{m+1}^{\bar{r}} \in \mathcal{U}^T}{\arg\min} \left|\left|\left| \bar{\bar{r}}_{m+1} \Delta\dot{\lambda}_{m+1}^{\bar{r}} - H_{\bar{R}} R_\infty \bar{\bar{r}}_{m+1} \Delta\lambda_{m+1}^{\bar{r}} + \Delta_{n+1}^{\bar{r}} \right|\right|\right|_{H_{\bar{R}}^{-1}}^2 \tag{7.25}$$

By discretizing the above expressions we obtain the following general equation (see chapter 4):

$$\left(\underline{\underline{\boldsymbol{Q}}}_{up} + \underline{\underline{\boldsymbol{\mathcal{L}}}}_{up} \right) \Delta\underline{\boldsymbol{\Lambda}}^{[k]} = \underline{\underline{\boldsymbol{\mathcal{R}}}}_{up} \Delta\underline{\boldsymbol{\Lambda}}^{[k-1]} + \underline{\boldsymbol{f}}_{up}^{[k]} \tag{7.26}$$

Using expression (7.26) and following the ideas as presented for the enrichment step, the temporal functions to be determined at the preliminary step can also be written for each time subdomain and micro iteration ℓ as:

$\forall \ell \in [1, 2],\ \forall k_p \in \left[0, ..., N_T^{(p)}\right],\ \forall p \in [1, 2, ..., N]$,

$$(\Delta\underline{\boldsymbol{\Lambda}})_{\ell,p}^{[k_p]} = \left[\left(\underline{\underline{\boldsymbol{Q}}}_{up} + \underline{\underline{\boldsymbol{\mathcal{L}}}}_{up} \right)^{-1} \underline{\underline{\boldsymbol{\mathcal{R}}}}_{up} \right]^{k_p} (\Delta\underline{\boldsymbol{\Lambda}})_{\ell,p}^{[0]} + \sum_{j=1}^{k_p} \left[\left(\underline{\underline{\boldsymbol{Q}}}_{up} + \underline{\underline{\boldsymbol{\mathcal{L}}}}_{up} \right)^{-1} \underline{\underline{\boldsymbol{\mathcal{R}}}}_{up} \right]^{k_p - j} \left(\underline{\underline{\boldsymbol{Q}}}_{up} + \underline{\underline{\boldsymbol{\mathcal{L}}}}_{up} \right)^{-1} (\underline{\boldsymbol{f}}_{up})_p^{[j]} \tag{7.27}$$

The above expression allows also a parallel resolution for the determination of the temporal corrective PGD functions, indeed all the procedures and equations presented for the enrichment step remain the same. In this sense, the macro and micro problems are solved in the same way as exposed previously, where the macro problem involves the use of the constant transmission operator $\underline{\underline{\boldsymbol{P}}}_{up}$ given by:

$$\underline{\underline{\boldsymbol{P}}}_{up} = \left[\left(\underline{\underline{\boldsymbol{Q}}}_{up} + \underline{\underline{\boldsymbol{\mathcal{L}}}}_{up} \right)^{-1} \underline{\underline{\boldsymbol{\mathcal{R}}}}_{up} \right]^{N_T^{(1)}},\ \underline{\underline{\boldsymbol{P}}}_{up} \in \mathbb{R}^{4m} \otimes \mathbb{R}^{4m} \tag{7.28}$$

As can be seen from the above developments, the parallel method presented here naturally arises from a classical temporal resolution using TDGM. The parallel strategy converges only in 3 stages, where the temporal function obtained corresponds exactly to a temporal function obtained by a serial solving strategy using the TDGM as presented in chapter 4. This is a strong point, since in this sense the parallel strategy does not modify the convergence rate of the LATIN-PGD method.

7.5 Numerical example

For the demonstration of the parallel strategy we consider herein the same example as the one treated in chapter 4 for the case of elastic-visco-plasticity, this choice of constitutive relation is totally arbitrary and the application of the parallel strategy for the case of isotropic damage or other complex material can be equally applicable with no difficulties. The bending test considered is recalled herein and is shown in figure 7.8.

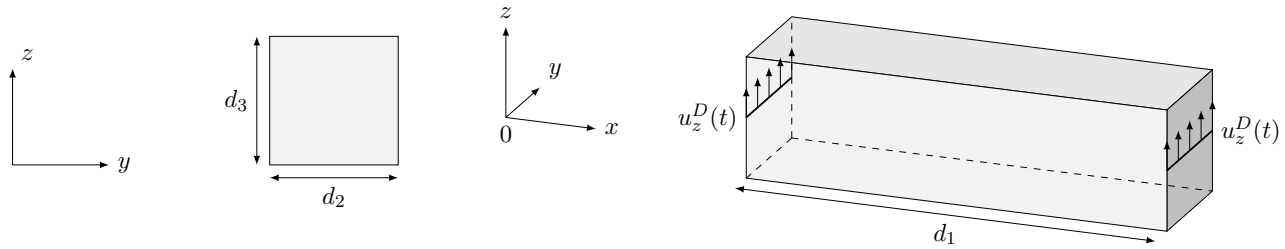


Figure 7.8: Test case considered, along with its dimensions.

We recall (as given in chapter 4) that the beam dimensions for this case are $d_1 = 6$ [m], $d_2 = d_3 = 0.1$ [m]. In addition, the vertical displacement $u_z^D(t)$ considered consists in a simple sinusoidal excitation of 3[Hz] as shown in figure 7.9.

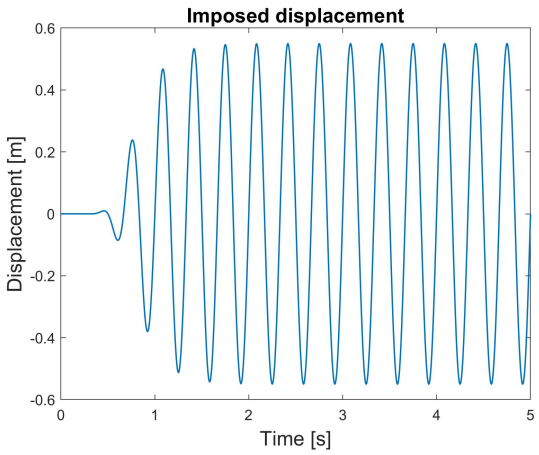


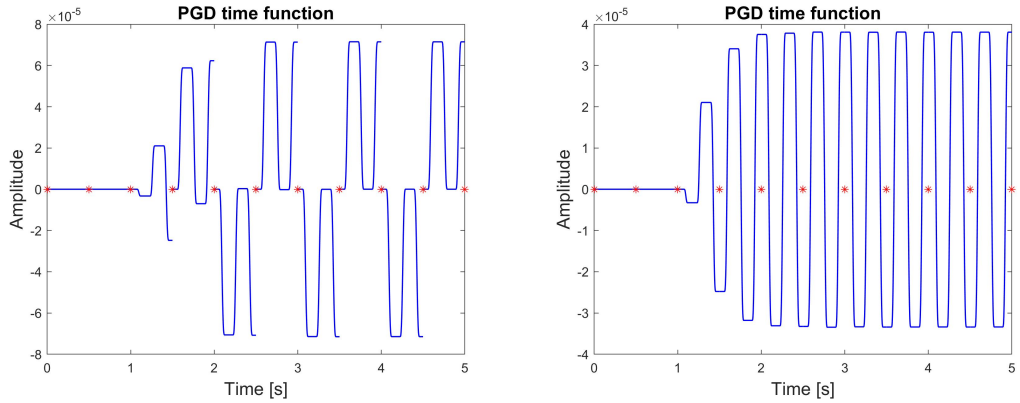
Figure 7.9: Imposed displacement.

From the results presented in chapter 4 the final solution is already known, so the principal idea of considering the same test is to demonstrate that exactly the same solution is obtained by the parallel strategy introduced in this chapter.

The results presented in this section were implemented in a serial code, so numerical savings are omitted, however, their correct implementation exploiting the parallel architecture of nowadays clusters is considered as a perspective for future developments.

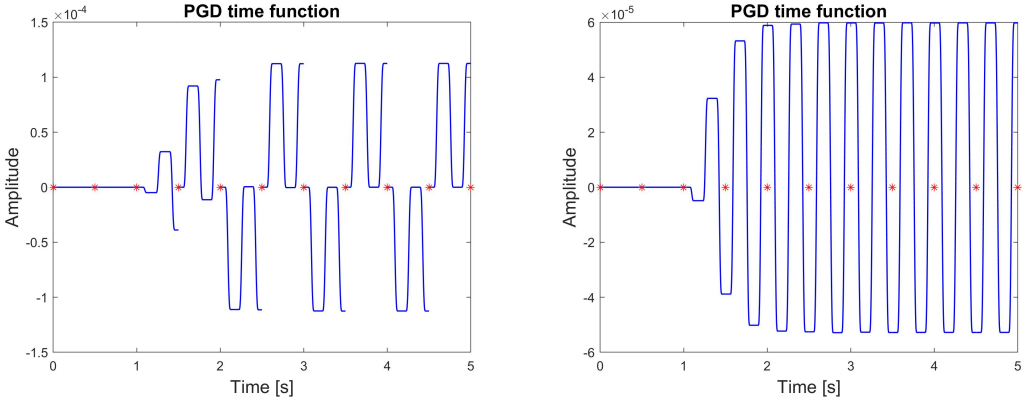
For the application of the parallel method 10 macro temporal elements are considered, this choice is completely arbitrary and its selection is based only for illustrative purposes. Under these considerations, figures 7.10, 7.11 and 7.12 illustrate the temporal PGD functions associated to the first rank of the PGD decomposition for the approximation of the global quantities associated to the admissibility conditions, the

kinematic and isotropic hardening respectively, where the macro temporal mesh is marked as red points. In these figures we can see the final results associated with the first and second micro problems.



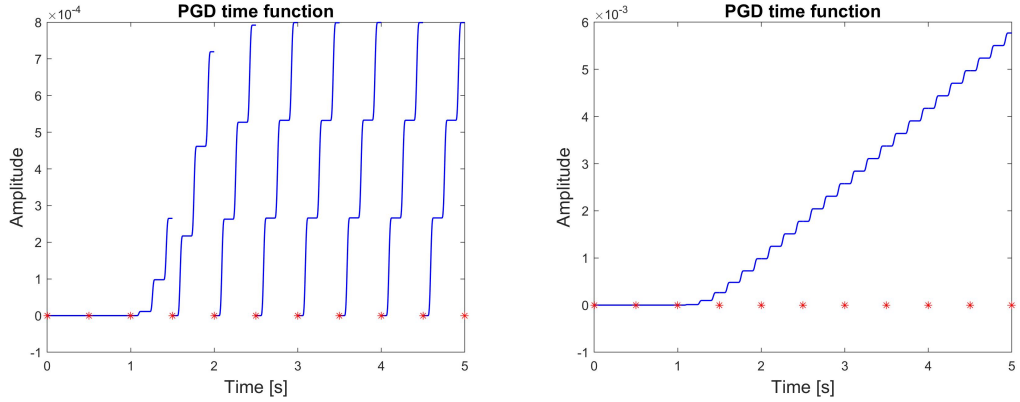
(a) First micro resolution. (b) Second micro resolution.

Figure 7.10: Enrichment step applied to the temporal PGD function $\lambda_1(t)$.



(a) First micro resolution. (b) Second micro resolution.

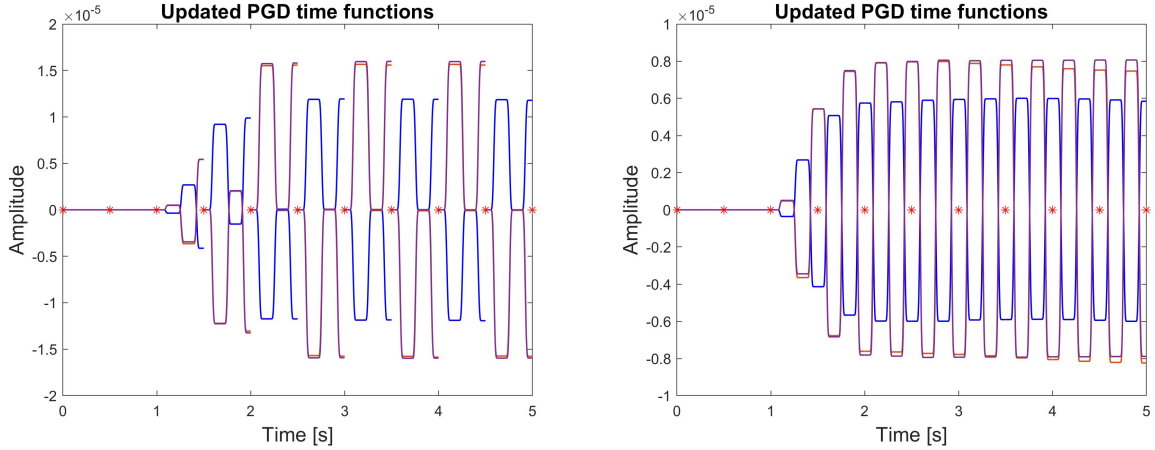
Figure 7.11: Enrichment step applied to the temporal PGD function $\lambda_1^{\alpha}(t)$.



(a) First micro resolution. (b) Second micro resolution.

Figure 7.12: Enrichment step applied to the temporal PGD function $\lambda_1^{\bar{r}}(t)$.

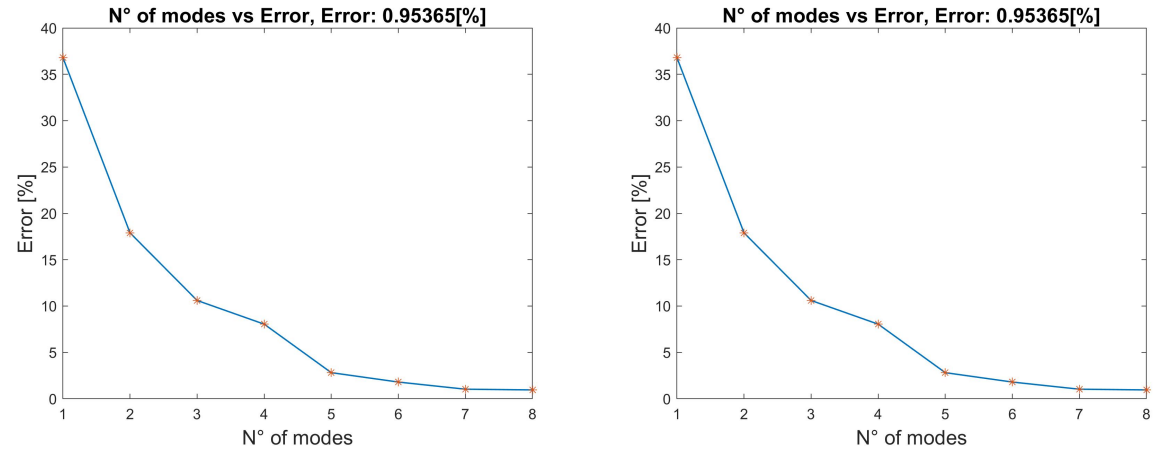
On the other hand, figures 7.13a and 7.13b show the results obtained for the first and second micro problems respectively at the preliminary step for the update of 3 time functions applied to the admissibility temporal terms (displacement, strain, plastic deformation and stress).



(a) First micro resolution. (b) Second micro resolution.

Figure 7.13: Preliminary step solutions when applying the parallel strategy for the update of 3 PGD temporal functions.

As mentioned previously, the parallel resolution strategy does not introduce approximation errors, so the convergence of the LATIN-PGD method remains the same as if a classical TDGM resolution strategy were used, this can be easily seen in figure 7.14 where the error curves versus the number of PGD modes are shown for the classical and parallel temporal strategy.



(a) Error using the classic strategy. (b) Error using the parallel strategy.

Figure 7.14: Error comparison between a classic and a parallel resolution strategy applied to the LATIN-PGD method.

The same convergence rate is obtained because the whole parallel solving strategy allows to resolve the same equations as if TDGM were used in a serial way as presented in chapter 4. This property is very powerful and outlines the advantages of the present strategy for future developments or its application to solve large industrial problems.

7.6 Conclusions

Model reduction methods are widely used in the context of solid mechanics when the reference problem contains a high amount of degrees of freedom and when calculating a direct resolution is prohibitive due to the high computational cost. But model reduction methods are not the only solution to reduce the computational burden, domain decomposition methods also does, which make use of the parallel architecture of current computers to distribute the load among them, obtaining as a consequence a considerable reduction of computational time.

In this context, this chapter introduced a parallel computational strategy to obtain the temporal functions of the LATIN-PGD decomposition. The main objective of this strategy is to achieve a greater reduction in computational time when the LATIN method and PGD are used, a reduction that increases with the number of processors used for the calculation. This strategy is based on the use of the Time Discontinuous Galerkin Method (TDGM) and the idea is to solve independently and in parallel the temporal PGD functions over temporal subdomains of uniform size. For this purpose, two "micro" and one "macro" problems are solved, where the micro problem consists in independently solving the time function on a given number of time macro elements, and whose calculation is performed independently in parallel. The "macro" problem is solved serially and its resolution is inexpensive due to the use of a "transmission operator" which solves the problem by performing inexpensive multiplicative operations. For summing things up, the technique starts by (i) setting all initial conditions on all subdomains equal to zero and solving the temporal functions independently on each of them (first micro problem), (ii) actualizing and correctly determining the initial values on each subdomain by solving a macro problem and finally (iii) a last micro problem is solved taking into account the correct initial conditions determined at the macro problem, where a continuous time function is obtained (in the sense of the TDGM). These three simple steps allow at the end of the process to obtain the desired temporal PGD functions, both in the enrichment step and in the updating of the time functions in the preliminary step. However, for the optimal application of the introduced parallel strategy, an uniform micro and macro discretization must be considered, along with constant operators related to the descent search direction at the global stage. These considerations allow the definition of constants "transmission operators" that drastically accelerate the resolution of the macro problems.

The parallel strategy introduced in this chapter does not require any modification of the spatial domain, since it only solves the temporal problems arising in the enrichment and preliminary steps. This is one of the main differences when compared to the pioneering works to achieve a parallel resolution in the LATIN framework [Ladevèze and Nouy, 2002, Ladevèze and Nouy, 2003, Nouy, 2003, Néron, 2004, Ladevèze et al., 2007, Passieux et al., 2008, Passieux et al., 2010, Ladevèze et al., 2010a, Ladevèze et al., 2010b]. In these references, a spatio-temporal homogenization technique is applied (see section 3.4.4), where a macro-micro spatio-temporal representation of the unknowns is introduced at the interfaces of the spatial subdomains. The temporal formulation in time follows ideas similar to the parallel strategy introduced in this chapter, especially for the micro problem, which simply consists of a resolution in each temporal subdomain. However, the macro problem differs, since in this case a time function is determined following a slow evolution in time and employing a FEM formulation.

Finally, a powerful property of the parallel strategy introduced in this chapter is the fact that no modification of the convergence properties is observed when compared to a classical solving strategy such as the one presented in chapter 4. This is because in both cases exactly the same temporal equations are solved in the enrichment and preliminary steps. Maintaining in this sense the same convergence rate in the LATIN-PGD method. This feature positions the parallel resolution method as a powerful technique for the efficient resolution of problems where the external excitation has a long duration and complex behavior.

Chapter 8

Conclusions and perspectives

The main objective of the present thesis work consisted on the development of dedicated strategies for the fast resolution of nonlinear low-frequency dynamics problems in solid mechanics, where the structure is subjected to loading with complex behavior and long duration in time. These developments considered the materials most commonly used for building construction, i.e. concrete and steel. From the above, two constitutive relationships, an isotropic damage model for the simulation of concrete materials and a standard version of the Marquis Chaboche elasto-visco-plasticity model for the simulations of metallic materials such as steel reinforcement, were considered. Under this context and due to the high computational cost when using incremental nonlinear solvers for their resolution, the LATIN method along with the model reduction technique *Proper Generalized Decomposition* (PGD) were considered. The LATIN method, contrary to incremental nonlinear approaches, solves the reference problem globally over the whole space and time domain at each iteration of the method. This global resolution strategy allows an easy introduction of model order reduction techniques into its formulation, which gives the LATIN-PGD its well-known efficiency. The use of the PGD allows to solve the nonlinear problem seeking the solution on a reduced order basis that can be calculated on the fly without previous knowledge of the solution. Despite the continuous development of the LATIN-PGD method in recent years, its adaptation for the treatment of problems in low-frequency dynamics has not been the focus of recent works until the present thesis, mainly because the attention had been centered on the study of problems under quasi-static conditions. In addition, the development of strategies dedicated to the rapid resolution of the time evolution of the LATIN method for the case of complex behavior and long time duration excitations, such as seismic or fatigue loads, is limited, although great efforts have been made in this area. For years, approximation techniques have been developed in the case of long-duration loading such as fatigue problems, however they are limited to the treatment of mono-periodic excitations [Cognard and Ladevèze, 1993, Arzt and Ladevèze, 1994, Bhattacharyya et al., 2018a, Bhattacharyya et al., 2018c, Bhattacharyya et al., 2019]. Along the same lines, domain decomposition and parallel computation methods have been introduced to deal with problems with a large number of degrees of freedom, both in space and time [Ladevèze and Nouy, 2003, Néron, 2004, Passieux, 2008], but its formulation is complex and require the sub-division of the spatial domain, which is not always desired.

In this sense, the need of solving low-frequency nonlinear dynamics problems with numerical efficiency, even in the case of external excitations of long duration and complex behavior, gave rise to the development of new solution strategies. Each of them aims at solving different points of great importance, these strategies are briefly explained below:

1. As mentioned above, one of the main objectives of the present thesis is to be able to solve low-frequency nonlinear dynamic problems, so the first development presented in chapter 4 consisted in the extension of the LATIN-PGD framework to take into account the dynamic effects of the struc-

ture. Additionally, in this chapter the Time Discontinuous Galerkin method is applied for a novel incremental resolution of the temporal PGD functions in both enrichment and preliminary steps. The LATIN-PGD method is verified for solving problems in dynamics by comparing the numerical results with a classical step-by-step solver (Newton-Raphson) for the case of isotropic damage and elasto-visco-plasticity.

2. Apart from the treatment of nonlinear problems in dynamics, the present thesis also focuses on the resolution of problems where the external load has a long time duration and complex behavior, such as seismic or fatigue loads. In this situation, the temporal degrees of freedom to be determined at each iteration of the LATIN-PGD can be very large, which would impair the efficiency of the solver. To circumvent this limitation, in chapter 5, a multiscale temporal approximation is developed, whose objective is to approximate the temporal PGD functions to be determined in the global stage of the LATIN-PGD method as a sum of different signal contributions whose computation is inexpensive and where the temporal degrees of freedom are drastically reduced.

3. As exposed in the previous point, when problems with a long time duration are studied, their discretization introduces a high number of temporal degrees of freedom. In this context, and due to the global resolution nature of the LATIN-PGD method, at each iteration of the method the processes of evaluating the constitutive relation at all spatio-temporal integration points at the local stage, as well as the spatio-temporal integration operations required at the global stage for the construction of the PGD approximation start to become very costly. Due to this problem, in chapter 6 a hyper-reduction technique is introduced. This method exploits the idea of spatio-temporal reference points to economically construct a low-rank approximation of the constitutive relation quantities in the whole spatio-temporal domain at the local stage, thus allowing a considerable reduction of the integration operations at the global stage and, consequently, drastically decreasing the computational cost of the whole LATIN-PGD method.

4. Finally, under the same objective of efficiently solving the time evolution of a nonlinear problem in the context of the LATIN-PGD method when long time duration external excitations are considered, in chapter 7 a parallel temporal resolution strategy is presented to take advantage of the current processor architecture. This resolution strategy allows obtaining in parallel the temporal PGD functions of both the enrichment and preliminary steps in the global stage, and its development arises naturally from the use of the Time Discontinuous Galerkin Method (TDGM).

Despite the number of developments that have been made during this doctoral work to improve the performance of the LATIN-PGD solver, there are still many interesting ideas to be tested, these ideas are presented below as perspectives for future research:

1. In this thesis work, all the operators related to the ascent search directions are considered to be constant. Therefore a first perspective could be the use of more optimized search directions by considering tangent operators in order to accelerate the convergence of the solver. Of course, this perspective excludes the parallel strategy presented in chapter 7, since the use of constant operators is a key requirement of the strategy.

2. In the context of the multiscale approach introduced in chapter 5, an interesting perspective could be the introduction of an iterative resolution method for the determination of the macro functions and the application of a parallel strategy for the determination of the micro functions in order to speed up the resolution. The use of the Time Discontinuous Galerkin method to accelerate the resolution of the macro functions is also considered as perspective.

3. The use of the hyper-reduction technique introduced in chapter 6 needs a constitutive relation that can be evaluated locally in the spatio-temporal domain such as the elasto-visco-plasticity constitutive relation. On the other hand, isotropic damage simulation for concrete material can be evaluated locally in the spatial domain but not in the time domain, on which a differential equation must be solved in time. The need to solve an ordinary differential equation in time for the case of isotropic damage behavior makes it impossible to apply the low-rank approximation of the hyper-reduced technique without calculating the complete spatio-temporal solution. This restriction makes necessary an extension of the method to take into account this type of constitutive relations. In this sense, an interesting perspective could be the approximation of the damage model by an explicit function that can be derived by applying Deep Learning techniques or other methods that can approximate the original behavior of the material. This mapping approach will require state-of-the-art developments and is considered a major challenge for the acceleration of the LATIN-PGD solver.

4. Finally, in the same line of the hyper-reduced technique, an interesting application arises naturally, which consists in the low-rank approximation of the variables of the constitutive relation over a spatio-temporal-parametric domain. This application seems very interesting when parametric studies have to be performed, especially when the constitutive relation has to be evaluated several times for each parameter. Thus, the use of the hyper-reduction technique presented in this thesis could drastically reduce the computational burden of such studies.

It should be emphasized that the ideas presented in the present thesis work were applied only to “academic” problems, on the one hand due to the prematurity of the new developments proposed, as well as the limited time of a thesis work. In this context, apart from the perspectives given previously for future research, as a general perspective it is considered the application of the different ideas developed in this thesis to solve real industrial problems, which can truly demonstrate the capabilities and optimizations presented in this thesis for the LATIN-PGD method.

It is humbly expected that the work carried out in this thesis will open the doors to new developments that further improve the performance of the LATIN-PGD method, that constitutes a favorable framework for including endless optimization techniques into its formulation that still remain to be developed.

The codes developed for the application of the different methods exposed in this thesis were carried out in MATLAB version R2020a, however these developments will be introduced later in CAST3M, which is an open source program developed by the CEA specially dedicated for numerical simulations, in order to allow its wide industrial usage.

Chapter 9

Appendix

Appendix A

Bilinear and linear discretization of the elastic dynamic problem

The following lines shows the construction of the bilinear ($\mathcal{A}_k(\underline{u}^{\mathcal{N}}, \underline{v}^{\mathcal{N}})$ and linear ($\mathcal{B}'_k(\underline{v}^{\mathcal{N}})$) operators for a given temporal element k (see chapter 2).

• **Bilinear operator construction**:

We start by calculating the operator $\mathcal{A}_k(\underline{u}^{\mathcal{N}}, \underline{v}^{\mathcal{N}})$ defined as follows:

$$\mathcal{A}_k(\underline{u}^{\mathcal{N}}, \underline{v}^{\mathcal{N}}) = \int_{\check{I}_k} \underline{\dot{\boldsymbol{v}}}(t) \cdot \left(\underline{\underline{\boldsymbol{M}}}\, \underline{\ddot{\boldsymbol{u}}}(t) + \underline{\underline{\boldsymbol{D}}}\, \underline{\dot{\boldsymbol{u}}}(t) + \underline{\underline{\boldsymbol{K}}}\, \underline{\boldsymbol{u}}(t) \right) \, dt \tag{A.1}$$

The discretization of the operator $\mathcal{A}_k$ by using the finite elements shape functions in space and time gives:

$$\begin{aligned} \int_{\check{I}_k} \underline{\dot{\boldsymbol{v}}}(t) \cdot \underline{\underline{\boldsymbol{M}}} \cdot \underline{\ddot{\boldsymbol{u}}}(t) dt &= \int_{\check{I}_k} \left(\sum_{j=1}^{4} \underline{\dot{\psi}}_j^{[k]}(t) \underline{\underline{\mathbf{v}}}_j^{[k]} \right) \cdot \underline{\underline{\boldsymbol{M}}} \cdot \left(\sum_{i=1}^{4} \underline{\ddot{\psi}}_i^{[k]}(t) \underline{\underline{\mathbf{u}}}_i^{[k]} \right) dt \\ &= \sum_{j=1}^{4} \sum_{i=1}^{4} \underline{\underline{\mathbf{v}}}_j^{[k]} \cdot \underline{\underline{\boldsymbol{M}}} \cdot \left(\int_{\check{I}_k} \underline{\dot{\psi}}_j^{[k]}(t) \underline{\ddot{\psi}}_i^{[k]}(t) dt \right) \underline{\underline{\mathbf{u}}}_i^{[k]} \\ &= \underline{\underline{\mathbf{v}}}^{[k]} : \left[\underline{\underline{\boldsymbol{M}}} \otimes \int_{\check{I}_k} \underline{\dot{\psi}}^{[k]}(t) \otimes \underline{\ddot{\psi}}^{[k]}(t) dt \right] : \underline{\underline{\mathbf{u}}}^{[k]} \end{aligned} \tag{A.2}$$

The discretized bilinear operator writes:

$$\mathcal{A}_k(\underline{u}^{\mathcal{N}}, \underline{v}^{\mathcal{N}}) = \underline{\underline{\mathbf{v}}}^{[k]} : \left[\underline{\underline{\boldsymbol{M}}} \otimes \underline{\underline{\boldsymbol{Q}}}_k^{12} + \underline{\underline{\boldsymbol{D}}} \otimes \underline{\underline{\boldsymbol{Q}}}_k^{11} + \underline{\underline{\boldsymbol{K}}} \otimes \underline{\underline{\boldsymbol{Q}}}_k^{10} \right] : \underline{\underline{\mathbf{u}}}^{[k]} \tag{A.3}$$

with:

$$\underline{\underline{\boldsymbol{Q}}}_k^{ij} = \int_{\check{I}_k} \frac{\partial^i \underline{\psi}^{[k]}(t)}{\partial t^i} \otimes \frac{\partial^j \underline{\psi}^{[k]}(t)}{\partial t^j} dt \tag{A.4}$$

By summing the continuity conditions between elements k and $k-1$ to the operator A_k, we can write:

$$\mathcal{A}'_k(\underline{u}^{\mathcal{N}}, \underline{v}^{\mathcal{N}}) = \mathcal{A}_k(\underline{u}^{\mathcal{N}}, \underline{v}^{\mathcal{N}}) + \underline{\dot{\boldsymbol{v}}}(t_{k-1}^+) \cdot \underline{\underline{\boldsymbol{M}}} \cdot \underline{\dot{\boldsymbol{u}}}(t_{k-1}^+) + \underline{\boldsymbol{v}}(t_{k-1}^+) \cdot \underline{\underline{\boldsymbol{K}}} \cdot \underline{\boldsymbol{u}}(t_{k-1}^+) \tag{A.5}$$

where the last two terms are given by:

$$\underline{\dot{\boldsymbol{v}}}(t_{k-1}^+) \cdot \underline{\underline{\boldsymbol{M}}} \cdot \underline{\dot{\boldsymbol{u}}}(t_{k-1}^+) = \underline{\underline{\mathbf{v}}}^{[k]} : \left[\underline{\underline{\boldsymbol{M}}} \otimes \left(\underline{\dot{\psi}}^{[k]}(t_{k-1}) \otimes \underline{\dot{\psi}}^{[k]}(t_{k-1}) \right) \right] : \underline{\underline{\mathbf{u}}}^{[k]} = \underline{\underline{\mathbf{v}}}^{[k]} : \left[\underline{\underline{\boldsymbol{M}}} \otimes \underline{\underline{\boldsymbol{P}}}_k^{11} \right] : \underline{\underline{\mathbf{u}}}^{[k]} \tag{A.6}$$

$$\underline{\boldsymbol{v}}(t_{k-1}^{+})\cdot\underline{\underline{\boldsymbol{K}}}\cdot\underline{\boldsymbol{u}}(t_{k-1}^{+})=\underline{\underline{\mathbf{v}}}^{[k]}:\left[\underline{\underline{\boldsymbol{K}}}\otimes\left(\underline{\psi}^{[k]}(t_{k-1})\otimes\underline{\psi}^{[k]}(t_{k-1})\right)\right]:\underline{\underline{\mathbf{u}}}^{[k]}=\underline{\underline{\mathbf{v}}}^{[k]}:\left[\underline{\underline{\boldsymbol{K}}}\otimes\underline{\underline{\boldsymbol{P}}}_{k}^{00}\right]:\underline{\underline{\mathbf{u}}}^{[k]} \tag{A.7}$$

with:

$$\underline{\underline{\boldsymbol{P}}}_{k}^{ij}=\frac{\partial^{i}\underline{\psi}^{[k]}(t_{k-1})}{\partial t^{i}}\otimes\frac{\partial^{j}\underline{\psi}^{[k]}(t_{k-1})}{\partial t^{j}} \tag{A.8}$$

• **Linear operator construction**:

The linear operator $\mathcal{B}_{k}'(\underline{v}^{\mathcal{N}})$ is given by :

$$\mathcal{B}_{k}'(\underline{v}^{\mathcal{N}})=\int_{\check{I}_{k}}\underline{\dot{\boldsymbol{v}}}(t)\cdot\underline{\boldsymbol{f}}(t)dt+\underline{\dot{\boldsymbol{v}}}(t_{k-1}^{+})\cdot\underline{\underline{\boldsymbol{M}}}\cdot\underline{\dot{\boldsymbol{u}}}(t_{k-1}^{-})+\underline{\boldsymbol{v}}(t_{k-1}^{+})\cdot\underline{\underline{\boldsymbol{K}}}\cdot\underline{\boldsymbol{u}}(t_{k-1}^{-}) \tag{A.9}$$

and the three discretized terms in time are calculated as follows:

$$\int_{\check{I}_{k}}\underline{\dot{\boldsymbol{v}}}(t)\cdot\underline{\boldsymbol{f}}(t)dt=\underline{\underline{\mathbf{v}}}^{[k]}:\left[\underline{\underline{\boldsymbol{I}}}\otimes\int_{\check{I}_{k}}\underline{\dot{\psi}}^{[k]}(t)\otimes\underline{\psi}^{[k]}(t)dt\right]:\underline{\underline{\mathbf{f}}}^{[k]}=\underline{\underline{\mathbf{v}}}^{[k]}:\left[\underline{\underline{\boldsymbol{I}}}\otimes\underline{\underline{\boldsymbol{Q}}}_{k}^{10}\right]:\underline{\underline{\mathbf{f}}}^{[k]} \tag{A.10}$$

$$\underline{\dot{\boldsymbol{v}}}(t_{k-1}^{+})\cdot\underline{\underline{\boldsymbol{M}}}\cdot\underline{\dot{\boldsymbol{u}}}(t_{k-1}^{-})=\underline{\underline{\mathbf{v}}}^{[k]}:\left[\underline{\underline{\boldsymbol{M}}}\otimes\left(\underline{\dot{\psi}}^{[k]}(t_{k-1})\otimes\underline{\dot{\psi}}^{[k-1]}(t_{k-1})\right)\right]:\underline{\underline{\mathbf{u}}}^{[k-1]}=\underline{\underline{\mathbf{v}}}^{[k]}:\left[\underline{\underline{\boldsymbol{M}}}\otimes\underline{\underline{\boldsymbol{R}}}_{k}^{11}\right]:\underline{\underline{\mathbf{u}}}^{[k-1]} \tag{A.11}$$

$$\underline{\boldsymbol{v}}(t_{k-1}^{+})\cdot\underline{\underline{\boldsymbol{K}}}\cdot\underline{\boldsymbol{u}}(t_{k-1}^{-})=\underline{\underline{\mathbf{v}}}^{[k]}:\left[\underline{\underline{\boldsymbol{K}}}\otimes\left(\underline{\psi}^{[k]}(t_{k-1})\otimes\underline{\psi}^{[k-1]}(t_{k-1})\right)\right]:\underline{\underline{\mathbf{u}}}^{[k-1]}=\underline{\underline{\mathbf{v}}}^{[k]}:\left[\underline{\underline{\boldsymbol{K}}}\otimes\underline{\underline{\boldsymbol{R}}}_{k}^{00}\right]:\underline{\underline{\mathbf{u}}}^{[k-1]} \tag{A.12}$$

with the following notation:

$$\underline{\underline{\boldsymbol{R}}}_{k}^{ij}=\frac{\partial^{i}\underline{\psi}^{[k]}(t_{k-1})}{\partial t^{i}}\otimes\frac{\partial^{j}\underline{\psi}^{[k-1]}(t_{k-1})}{\partial t^{j}} \tag{A.13}$$

• **Taking into account the initial conditions**:

The information of the initial condition of the problem is taken into account in the linear operator $\mathcal{B}_{1}'(.)$, that is:

$$\mathcal{B}_{1}'(\underline{v}^{\mathcal{N}})=\int_{\check{I}_{1}}\underline{\dot{\boldsymbol{v}}}(t)\cdot\underline{\boldsymbol{f}}(t)dt+\underline{\dot{\boldsymbol{v}}}(t_{0}^{+})\cdot\underline{\underline{\boldsymbol{M}}}\cdot\underline{\dot{\boldsymbol{u}}}_{in}+\underline{\boldsymbol{v}}(t_{0}^{+})\cdot\underline{\underline{\boldsymbol{K}}}\cdot\underline{\boldsymbol{u}}_{in} \tag{A.14}$$

with $\underline{\boldsymbol{u}}_{in}$ and $\underline{\dot{\boldsymbol{u}}}_{in}$ the initial displacement and velocity in $I=[0,T]$. This term is calculated as:

$$\int_{\check{I}_{1}}\underline{\dot{\boldsymbol{v}}}(t)\cdot\underline{\boldsymbol{f}}(t)dt=\underline{\underline{\mathbf{v}}}^{[1]}:\left[\underline{\underline{\boldsymbol{I}}}\otimes\int_{\check{I}_{1}}\underline{\dot{\psi}}^{[1]}(t)\otimes\underline{\psi}^{[1]}(t)dt\right]:\underline{\underline{\mathbf{f}}}^{[1]}=\underline{\underline{\mathbf{v}}}^{[1]}:\left[\underline{\underline{\boldsymbol{I}}}\otimes\underline{\underline{\boldsymbol{Q}}}_{1}^{10}\right]:\underline{\underline{\mathbf{f}}}^{[1]} \tag{A.15}$$

$$\underline{\dot{\boldsymbol{v}}}(t_{0}^{+})\cdot\underline{\underline{\boldsymbol{M}}}\cdot\underline{\dot{\boldsymbol{u}}}_{in}=\underline{\underline{\mathbf{v}}}^{[1]}:\left[\underline{\underline{\boldsymbol{M}}}\cdot\underline{\dot{\boldsymbol{u}}}_{in}\otimes\underline{\dot{\psi}}^{[1]}(t_{0})\right]=\underline{\underline{\mathbf{v}}}^{[1]}:\left[\underline{\underline{\boldsymbol{M}}}\cdot\underline{\dot{\boldsymbol{u}}}_{in}\otimes\underline{\dot{\psi}}^{[1]}(0)\right] \tag{A.16}$$

$$\underline{\boldsymbol{v}}(t_{0}^{+})\cdot\underline{\underline{\boldsymbol{K}}}\cdot\underline{\boldsymbol{u}}_{in}=\underline{\underline{\mathbf{v}}}^{[1]}:\left[\underline{\underline{\boldsymbol{K}}}\cdot\underline{\boldsymbol{u}}_{in}\otimes\underline{\psi}^{[1]}(t_{0})\right]=\underline{\underline{\mathbf{v}}}^{[1]}:\left[\underline{\underline{\boldsymbol{K}}}\cdot\underline{\boldsymbol{u}}_{in}\otimes\underline{\psi}^{[1]}(0)\right] \tag{A.17}$$

• **Final assembly**: Solving the equilibrium equation in a discretized weak sense leads to compute the discrete tensor $\underline{\underline{\mathbf{u}}}^{[k]}$ (on intervals $\check{I}_{k}$) as the solution of:

$$\begin{aligned}\left[\underline{\underline{\boldsymbol{M}}}\otimes(\underline{\underline{\boldsymbol{Q}}}_{k}^{12}+\underline{\underline{\boldsymbol{P}}}_{k}^{11})+\underline{\underline{\boldsymbol{D}}}\otimes\underline{\underline{\boldsymbol{Q}}}_{k}^{11}+\underline{\underline{\boldsymbol{K}}}\otimes(\underline{\underline{\boldsymbol{Q}}}_{k}^{10}+\underline{\underline{\boldsymbol{P}}}_{k}^{00})\right]:\underline{\underline{\mathbf{u}}}^{[k]}=\\ (\underline{\underline{\boldsymbol{I}}}\otimes\underline{\underline{\boldsymbol{Q}}}_{k}^{10}):\underline{\underline{\mathbf{f}}}^{[k]}+\left[\underline{\underline{\boldsymbol{M}}}\otimes\underline{\underline{\boldsymbol{R}}}_{k}^{11}+\underline{\underline{\boldsymbol{K}}}\otimes\underline{\underline{\boldsymbol{R}}}_{k}^{00}\right]:\underline{\underline{\mathbf{u}}}^{[k-1]}\end{aligned} \tag{A.18}$$

The discrete solution $\underline{\underline{\mathbf{u}}}^{[1]}$ over the initial element $\breve{I}_1$ in turn verifies:

$$\left[\underline{\underline{\boldsymbol{M}}} \otimes (\underline{\underline{\boldsymbol{Q}}}_1^{12} + \underline{\underline{\boldsymbol{P}}}_1^{11}) + \underline{\underline{\boldsymbol{D}}} \otimes \underline{\underline{\boldsymbol{Q}}}_1^{11} + \underline{\underline{\boldsymbol{K}}} \otimes (\underline{\underline{\boldsymbol{Q}}}_1^{10} + \underline{\underline{\boldsymbol{P}}}_1^{00})\right] : \underline{\underline{\mathbf{u}}}^{[1]} = (\underline{\underline{\boldsymbol{I}}} \otimes \underline{\underline{\boldsymbol{Q}}}_1^{10}) : \underline{\underline{\mathbf{f}}}^{[1]} + (\underline{\underline{\boldsymbol{M}}} \cdot \dot{\underline{\boldsymbol{u}}}_{in}) \otimes \underline{\boldsymbol{P}}_1^1 + (\underline{\underline{\boldsymbol{K}}} \cdot \underline{\boldsymbol{u}}_{in}) \otimes \underline{\boldsymbol{P}}_1^0 \tag{A.19}$$

with:

$$\underline{\boldsymbol{P}}_1^1 = \underline{\psi}^{[1]}(0) \tag{A.20}$$

$$\underline{\boldsymbol{P}}_1^0 = \dot{\underline{\psi}}^{[1]}(0) \tag{A.21}$$

Appendix B

General framework for the incremental resolution of nonlinear solid mechanics problems

Lets consider a classic incremental strategy for which a solution $s_k = \{\underline{u}_k, \underline{\underline{\varepsilon}}_k, \underline{\underline{\sigma}}_k\}$ is computed at time step t_k and where the solution $s_{k+1} = \{\underline{u}_{k+1}, \underline{\underline{\varepsilon}}_{k+1}, \underline{\underline{\sigma}}_{k+1}\}$ at next time step t_{k+1} is sought, given the loading terms $\underline{f}_{k+1} \in \Omega, \underline{f}^N_{k+1} \in \partial_N\Omega$ and $\underline{u}^D_{k+1} \in \partial_D\Omega$. The resolution of the problem mainly consists on two steps, that are:

1. The local integration of the nonlinear constitutive relations that gives $\underline{\underline{\sigma}}_{k+1}$ in function of $\underline{u}_{k+1}$ and s_k.
2. The linearization of the global equilibrium equations of the system and their resolution in an iterative way.

The global equilibrium of the structure in space at current time step t_{k+1} can be written as:
$\forall \underline{v} \in \mathcal{U}^S(\Omega, 0)$,

$$\int_\Omega \rho \ddot{\underline{u}}_{k+1} \cdot \underline{v} d\Omega + \int_\Omega \underline{\underline{\sigma}}_{k+1} : \underline{\underline{\varepsilon}}(\underline{v}) \, d\Omega = \int_\Omega \underline{f}_{k+1} \cdot \underline{v} \, d\Omega + \int_{\partial_N\Omega} \underline{f}^N_{k+1} \cdot \underline{v} \, dS \tag{B.1}$$

where the nonlinear stress tensor $\underline{\underline{\sigma}}_{k+1}$ writes as a function of the solution s_k at previous time-step and the displacement increment $\Delta\underline{u}_k = \underline{u}_{k+1} - \underline{u}_k$ writing:

$$\underline{\underline{\sigma}}_{k+1} = \mathcal{J}(\underline{\underline{\varepsilon}}(\Delta\underline{u}_k); s_k) \tag{B.2}$$

B.1 Temporal incremental approximation

In dynamics, the Newmark method is generally used for the incremental approximation in time [Dokainish and Subbaraj, 1989, Gavin, 2001]. In the Newmark framework the temporal evolution of the displacement is approximated as:

$$\underline{u}_{k+1} \approx \underline{u}_k + \Delta t \dot{\underline{u}}_k + \Delta t^2 \left[\left(\frac{1}{2} - \beta_s \right) \ddot{\underline{u}}_k + \beta_s \ddot{\underline{u}}_{k+1} \right] \tag{B.3}$$

$$\dot{\underline{u}}_{k+1} \approx \dot{\underline{u}}_k + \Delta t \left[(1 - \gamma_s) \ddot{\underline{u}}_k + \gamma_s \ddot{\underline{u}}_{k+1} \right] \tag{B.4}$$

where the corrections in terms of velocity $\delta\underline{\dot{u}}_k = \underline{\dot{u}}_{k+1} - \underline{\dot{u}}_k$ and acceleration $\delta\underline{\ddot{u}}_k = \underline{\ddot{u}}_{k+1} - \underline{\ddot{u}}_k$ are in function of the displacement correction $\delta\underline{u}_k = \underline{u}_{k+1} - \underline{u}_k$ as:

$$\delta\underline{\dot{u}}_k = \frac{\gamma_s}{\beta_s \Delta t}\delta\underline{u}_k - \frac{\gamma_s}{\beta_s}\underline{\dot{u}}_k + \Delta t\left(1 - \frac{\gamma_s}{2\beta_s}\right)\underline{\ddot{u}}_k \tag{B.5}$$

$$\delta\underline{\ddot{u}}_k = \frac{1}{\beta_s \Delta t^2}\delta\underline{u}_k - \frac{1}{\beta_s \Delta t}\underline{\dot{u}}_k - \frac{1}{2\beta_s}\underline{\ddot{u}}_k \tag{B.6}$$

From equations (B.3) and (B.4) one can write the acceleration as:

$$\underline{\ddot{u}}_{k+1} = \mathcal{L}(\delta\underline{u}_k; s_k) \tag{B.7}$$

which its full expression is given by:

$$\underline{\ddot{u}}_{k+1} \approx \frac{\delta\underline{u}_k - \Delta t\underline{\dot{u}}_k - \Delta t^2\left(\frac{1}{2} - \beta_s\right)\underline{\ddot{u}}_k}{\Delta t^2 \beta_s} \tag{B.8}$$

B.2 Incremental resolution

For each time step t_{k+1}, the nonlinear problem consists in finding a kinematically admissible correction $\Delta\underline{u}_k$ verifying equilibrium (B.1), constitutive relations (B.2) and acceleration prediction (B.8). Hence, the resolution of the global equilibrium consists in finding an incremental $\Delta\underline{u}_k \in \mathcal{U}^S(\Omega, \Delta\underline{u}_k^D)$ such as:

$\forall \underline{v} \in \mathcal{U}^S(\Omega, 0)$,

$$\mathcal{R}(\Delta\underline{u}_k; \underline{v}, s_k) = 0 \tag{B.9}$$

where the dynamic residual equilibrium writes:

$\forall \underline{v} \in \mathcal{U}^S(\Omega, 0)$,

$$\mathcal{R}(\Delta\underline{u}_k; \underline{v}, s_k) = \int_\Omega \rho\mathcal{L}(\Delta\underline{u}_k; s_k)\cdot\underline{v}d\Omega + \int_\Omega \mathcal{J}(\underline{\underline{\varepsilon}}(\Delta\underline{u}_k; s_k) : \underline{\underline{\varepsilon}}(\underline{v})d\Omega - \int_\Omega \underline{f}_{k+1}\cdot\underline{v}\, d\Omega - \int_{\partial_N\Omega}\underline{f}_{k+1}^N\cdot\underline{v}\, dS \tag{B.10}$$

The residue of equation (B.10) is minimized by approaching the incremental displacement $\Delta\underline{u}_k$ as the sum of $n+1$ incremental corrections, this is:

$$\Delta\underline{u}_k \approx \Delta\underline{u}_k^{(n+1)} = \Delta\underline{u}_k^{(n)} + \delta\underline{u}_k^{(n)} \tag{B.11}$$

where each incremental correction $\delta\underline{u}_k^{(n)}$ is solution of the equation (B.10) linearized around $\Delta\underline{u}_k^{(n)}$ in a Newton-Raphson manner, as illustrated in figure B.1, which is written:

$\forall \underline{v} \in \mathcal{U}^S(\Omega, 0)$,

$$\mathcal{R}'(\Delta\underline{u}_k^{(n)}; \underline{v}, s_k)\cdot\delta\underline{u}_k^{(n)} = -\mathcal{R}(\Delta\underline{u}_k^{(n)}; \underline{v}, s_k) \tag{B.12}$$

The functional $\mathcal{R}' = \frac{\partial\mathcal{R}}{\partial\Delta\underline{u}_k^{(n)}}$ corresponds to the tangent linear application of the residue $\mathcal{R}$ evaluated at $\Delta\underline{u}_k^{(n)}$:

$$\begin{aligned}\mathcal{R}'(\Delta\underline{u}_k^{(n)}; \underline{v}, s_k)\cdot\delta\underline{u}_k^{(n)} &= \int_\Omega \rho\delta\underline{u}_k^{(n)}\mathcal{L}'(\Delta\underline{u}_k; s_k)\cdot\underline{v}d\Omega + \int_\Omega \mathcal{J}'(\underline{\underline{\varepsilon}}(\Delta\underline{u}_k); s_k) : \underline{\underline{\varepsilon}}(\delta\underline{u}_k^{(n)}) : \underline{\underline{\varepsilon}}(\underline{v})d\Omega \\ &= \int_\Omega \rho\delta\underline{u}_k^{(n)}\left(\frac{1}{\beta_s\Delta t^2}\right)\cdot\underline{v}d\Omega + \int_\Omega \mathbb{T}_{op}(\Delta\underline{\underline{\varepsilon}}_k^{(n)}; s_k) : \underline{\underline{\varepsilon}}(\delta\underline{u}_k^{(n)}) : \underline{\underline{\varepsilon}}(\underline{v})d\Omega\end{aligned} \tag{B.13}$$

where $\mathbb{T}_{op}$ is the tangent operator that can be expressed as:

$$\mathbb{T}_{op}(\Delta\underline{\underline{\varepsilon}}_k^{(n)}; s_k) = \frac{\partial \underline{\underline{\sigma}}_{k+1}}{\partial \Delta\underline{\underline{\varepsilon}}_k}(\Delta\underline{\underline{\varepsilon}}_k^{n}; s_k) = \mathbb{K}'(\Delta\underline{\underline{\varepsilon}}_k^{(n)}; s_k) + \left(\frac{\gamma_s}{\beta_s \Delta t}\right)\mathbb{D}'(\Delta\underline{\underline{\varepsilon}}_k^{(n)}; s_k) \tag{B.14}$$

introducing $\mathbb{K}'$ and $\mathbb{D}'$ the tangent stiffness and damping operators.

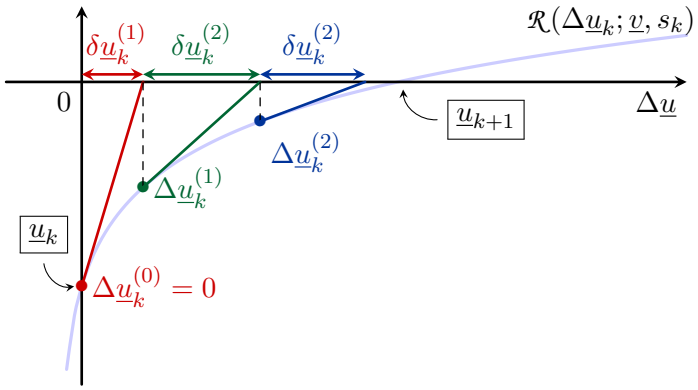


Figure B.1: Calculation of $\Delta\underline{u}_k$ as a sum of incremental corrections $\delta\underline{u}_k^n$.

B.2.1 Spatial discretization and temporal incremental resolution

After discretizing the space domain $\mathcal{U}^S$ by using finite elements in space, the solution of equation (B.1) is searched in the subspace $\mathcal{U}_h^S$ of dimension n_S. Therefore the vectors of nodal values, the residual at time t_{k+1} writes:

$\forall \underline{v}^h \in \mathcal{U}_h^S(\Omega, 0),\ \forall \underline{\boldsymbol{v}} \in \mathbb{R}^{n_S}$,

$$\underbrace{\mathcal{R}(\Delta\underline{u}_k^h; \underline{v}^h, s_k)}_{\underline{\boldsymbol{v}}^T \underline{\boldsymbol{R}}_{k+1}} = \underbrace{\int_\Omega \rho \underline{\ddot{u}}_{k+1}^h \cdot \underline{v}^h \, d\Omega}_{\underline{\boldsymbol{v}}^T \underline{\underline{\boldsymbol{M}}}\, \underline{\ddot{\boldsymbol{u}}}_{k+1}} + \underbrace{\int_\Omega \underline{\underline{\sigma}}_{k+1} : \underline{\underline{\varepsilon}}(\underline{v}^h) \, d\Omega}_{\underline{\boldsymbol{v}}^T \underline{\boldsymbol{g}}_{k+1}} \underbrace{- \int_\Omega \underline{f}_{k+1} \cdot \underline{v}^h \, d\Omega - \int_{\partial\Omega_N} \underline{f}_{k+1}^N \cdot \underline{v}^h \, dS}_{-\underline{\boldsymbol{v}}^T \underline{\boldsymbol{f}}_{k+1}} \tag{B.15}$$

where we have the inertial loading $\underline{\underline{\boldsymbol{M}}}\, \underline{\ddot{\boldsymbol{u}}}_{k+1} \in \mathbb{R}^{n_S}$, the vector $\underline{\boldsymbol{g}}_{k+1} \in \mathbb{R}^{n_S}$ of internal forces and the vector $\underline{\boldsymbol{f}}_{k+1} \in \mathbb{R}^{n_S}$ of external forces.

A discretized version of the tangent operator of the functional $\mathcal{R}$ can be introduced writing:

$$\mathcal{R}'(\Delta\underline{u}_k^{(n)}; \underline{v}, s_k) \cdot \delta\underline{u}_k^{(n)} \approx \underline{\underline{\boldsymbol{S}}}_k \, \delta\underline{\boldsymbol{u}}_k^{(n)}$$

where the tangent operator writes in function of the chosen β_s and γ_s parameters of the Newmark scheme:

$$\underline{\underline{\boldsymbol{S}}}_k = \underline{\underline{\boldsymbol{K}}}'_k + \frac{\gamma_s}{\beta_s \Delta t}\underline{\underline{\boldsymbol{D}}}'_k + \frac{1}{\beta_s \Delta t^2}\underline{\underline{\boldsymbol{M}}} \tag{B.16}$$

denoting $\underline{\underline{\boldsymbol{K}}}'_k$ and $\underline{\underline{\boldsymbol{D}}}'_k$ the tangent matrices that have to be recomputed at each time step k. Alternatively, those matrices can be chosen constant (quasi-Newton algorithm) $\underline{\underline{\boldsymbol{K}}}'_k = \underline{\underline{\boldsymbol{K}}}$ and $\underline{\underline{\boldsymbol{D}}}'_k = \underline{\underline{\boldsymbol{D}}}$; the convergence might be slower considering the iterations count, but time is saved since the re-computation and reassembling (time consuming step) of the tangent matrices are no longer needed. Here we assume a quasi-Newton algorithm, therefore $\underline{\underline{\boldsymbol{S}}}_k = \underline{\underline{\boldsymbol{S}}}$.

The corrections are computed as follows:

- **Iteration** $n = 1$, application of Dirichlet conditions:

  The first correction is searched on the space $\mathcal{U}^S(\Omega, \Delta\underline{u}_k^D)$, in a way that $\Delta\underline{u}_k$ is admissible to $\Delta\underline{u}_k^D$:

$$\delta\underline{u}_k^{(1)} = \delta\underline{u}_k^{(1,0)} + \Delta\underline{u}_k^D \tag{B.17}$$

  where $\Delta\underline{u}_k^D$ corresponds to the increment from time t_k to t_{k+1} of the imposed displacement. The iterations for $n > 1$ are therefore searched on the space $\mathcal{U}^S(\Omega, 0)$, so that $\Delta\underline{u}_k^{(n)}$, by construction, will be kinematically admissible to $\Delta\underline{u}_k^D$. The determination of $\delta\underline{u}_k^{(1,0)}$ is given by:

$$\mathcal{R}'(0; \underline{v}, s_k) \cdot \delta\underline{u}_k^{(1,0)} = -\mathcal{R}(0; \underline{v}, s_k) - \mathcal{R}'(0; \underline{v}, s_k) \cdot \Delta\underline{u}_k^D \tag{B.18}$$

  which, after discretization, consists in finding $\delta\underline{\boldsymbol{u}}_k^{(1,0)}$ verifying:

$$\underline{\underline{\boldsymbol{S}}}\, \delta\underline{\boldsymbol{u}}_k^{(1,0)} = \underbrace{-\underline{\underline{\boldsymbol{M}}}\, \underline{\ddot{\boldsymbol{u}}}_{k+1}^* - \underline{\boldsymbol{g}}_k + \underline{\boldsymbol{f}}_{k+1} - \underline{\underline{\boldsymbol{S}}}^D\, \Delta\underline{\boldsymbol{u}}_k^D}_{\underline{\boldsymbol{R}}_{k+1}^{(1)}} \tag{B.19}$$

  where $\underline{\underline{\boldsymbol{S}}}^D$ is the tangent operator addressed on the degrees of freedoms related to the imposed displacements. The inertial guess term at t_{k+1} is calculated by using equation (B.8) and setting $\Delta\underline{\boldsymbol{u}}_k = 0$, giving:

$$\underline{\ddot{\boldsymbol{u}}}_{k+1}^* \approx \frac{-\Delta t \underline{\dot{\boldsymbol{u}}}_k - \Delta t^2 \left(\frac{1}{2} - \beta_s\right) \underline{\ddot{\boldsymbol{u}}}_k}{\Delta t^2 \beta_s} \tag{B.20}$$

- **Iteration** $n > 1$:

  For $(n > 1)$ we seek a correction $\delta\underline{\boldsymbol{u}}_k^{(n)} \in \mathbb{R}^{n_S}$ verifying:

$$\underline{\underline{\boldsymbol{S}}}\, \delta\underline{\boldsymbol{u}}_k^{(n)} = \underbrace{-\underline{\underline{\boldsymbol{M}}}\, \underline{\ddot{\boldsymbol{u}}}_{k+1}^{(n)} - \underline{\boldsymbol{g}}_{k+1}^{(n)} + \underline{\boldsymbol{f}}_{k+1}}_{\underline{\boldsymbol{R}}_{k+1}^{(n)}} \tag{B.21}$$

Once the correction in displacement determined, the correction in speed and acceleration are computed using expressions (B.5) and (B.6) respectively.

The iterative procedure is applied until a given error threshold is reached, which is given as follows:

$$\left\|\underline{\boldsymbol{R}}_{k+1}\right\| < e \tag{B.22}$$

with $\|\cdot\|$ the euclidean norm and e a given number. The error defined in (B.22) measures the verification of the equilibrium equation.

Appendix C

Enrichment step: temporal problem of internal variables by using TDGM

In the following lines the details related to the calculation of the temporal functions at the enrichment step for the case of the internal variables is detailed. Lets consider for this the following simplification for the notations (the index related to the PGD mode is dropped):

$$\begin{aligned}\Delta\dot{\underline{\underline{\alpha}}}_{n+1} &= \underline{\underline{\bar{\alpha}}}_{m+1}(\underline{x})\dot{\lambda}^{\alpha}_{m+1}(t) = \underline{\underline{\bar{\alpha}}}(\underline{x})\dot{\lambda}^{\alpha}(t)\\ \Delta\dot{\bar{r}}_{n+1} &= \bar{\bar{r}}_{m+1}(\underline{x})\dot{\lambda}^{\bar{r}}_{m+1}(t) = \bar{\bar{r}}(\underline{x})\dot{\lambda}^{\bar{r}}(t)\end{aligned} \tag{C.1}$$

The temporal functions are determined such as they minimize their respective constitutive relations errors, this means:

$$\{\lambda^{\alpha}\} = \underset{\lambda^{\alpha}\in\mathcal{U}^T}{\arg\min} \left|\left|\left|\underline{\underline{\bar{\alpha}}}\dot{\lambda}^{\alpha} + \mathbb{H}_{\beta} : C\underline{\underline{\bar{\alpha}}}\lambda^{\alpha} + \underline{\underline{\Delta}}^{\alpha}_{n+1}\right|\right|\right|^2_{\mathbb{H}^{-1}_{\beta}} \tag{C.2}$$

$$\{\lambda^{\bar{r}}\} = \underset{\lambda^{\bar{r}}\in\mathcal{U}^T}{\arg\min} \left|\left|\left|\bar{\bar{r}}\dot{\lambda}^{\bar{r}} + H_{\bar{R}}R_{\infty}\bar{\bar{r}}\lambda^{\bar{r}} + \Delta^{\bar{r}}_{n+1}\right|\right|\right|^2_{H^{-1}_{\bar{R}}} \tag{C.3}$$

By developing the minimization of the equation (C.2) and (C.3) we obtain respectively:
$\forall \delta\lambda^{\alpha}, \delta\lambda^{\bar{r}} \in \mathcal{U}^T$,

$$\int_{\Omega\times I}\left(\underline{\underline{\bar{\alpha}}}\delta\dot{\lambda}^{\alpha} + \mathbb{H}_{\beta} : C\underline{\underline{\bar{\alpha}}}\delta\lambda^{\alpha}\right) : \mathbb{H}^{-1}_{\beta} : \left(\underline{\underline{\bar{\alpha}}}\dot{\lambda}^{\alpha} + \mathbb{H}_{\beta} : C\underline{\underline{\bar{\alpha}}}\lambda^{\alpha} + \underline{\underline{\Delta}}^{\alpha}\right) d\Omega dt = 0 \tag{C.4}$$

$$\int_{\Omega\times I}\left(\bar{\bar{r}}\delta\dot{\lambda}^{\bar{r}} + H_{\bar{R}}R_{\infty}\bar{\bar{r}}\delta\lambda^{\bar{r}}\right) : H^{-1}_{\bar{R}} : \left(\bar{\bar{r}}\dot{\lambda}^{\bar{r}} + H_{\bar{R}}R_{\infty}\bar{\bar{r}}\lambda^{\bar{r}} + \Delta^{\bar{r}}\right) d\Omega dt = 0 \tag{C.5}$$

By defining the constants and temporal functions as follows:

$$\begin{aligned} A^{11,\alpha} &= \int_{\Omega}\underline{\underline{\bar{\alpha}}} : \mathbb{H}^{-1}_{\beta} : \underline{\underline{\bar{\alpha}}}\, d\Omega \\ A^{10,\alpha} &= \int_{\Omega} C\underline{\underline{\bar{\alpha}}} : \underline{\underline{\bar{\alpha}}}\, d\Omega \qquad\qquad D^{1,\alpha}(t) = -\int_{\Omega}\underline{\underline{\bar{\alpha}}} : \mathbb{H}^{-1}_{\beta} : \underline{\underline{\Delta}}^{\alpha}\, d\Omega \\ A^{01,\alpha} &= \int_{\Omega} C\underline{\underline{\bar{\alpha}}} : \underline{\underline{\bar{\alpha}}}\, d\Omega \qquad , \qquad D^{0,\alpha}(t) = -\int_{\Omega} C\underline{\underline{\bar{\alpha}}} : \underline{\underline{\Delta}}^{\alpha}\, d\Omega \\ A^{00,\alpha} &= \int_{\Omega} C^2\mathbb{H}_{\beta} : \underline{\underline{\bar{\alpha}}} : \underline{\underline{\bar{\alpha}}}\, d\Omega \end{aligned} \tag{C.6}$$

and:

$$\begin{aligned} A^{11,\bar{r}} &= \int_\Omega \bar{\bar{r}}\, H_{\bar{R}}^{-1} \bar{\bar{r}}\, d\Omega \\ A^{10,\bar{r}} &= \int_\Omega R_\infty \bar{\bar{r}}^2\, d\Omega \qquad\qquad D^{1,\bar{r}}(t) = -\int_\Omega \bar{\bar{r}}\, H_{\bar{R}}^{-1} \Delta^{\bar{r}}\, d\Omega \\ A^{01,\bar{r}} &= \int_\Omega R_\infty \bar{\bar{r}}^2\, d\Omega \qquad , \qquad D^{0,\bar{r}}(t) = -\int_\Omega R_\infty \bar{\bar{r}} \Delta^{\bar{r}}\, d\Omega \\ A^{00,\bar{r}} &= \int_\Omega R_\infty^2 H_{\bar{R}} \bar{\bar{r}}^2\, d\Omega \end{aligned} \tag{C.7}$$

By discretizing the equations (C.4), (C.5) and using (C.6), (C.7) we obtain the followings elemental matrices and vector at temporal element $[k]$:

$$\underline{\underline{\boldsymbol{Q}}}^{[k],\alpha} = \int_{\breve{I}_k} \left(\dot{\underline{\psi}}^{[k]} \otimes \dot{\underline{\psi}}^{[k]} A^{11,\alpha} + \dot{\underline{\psi}}^{[k]} \otimes \underline{\psi}^{[k]} A^{10,\alpha} + \underline{\psi}^{[k]} \otimes \dot{\underline{\psi}}^{[k]} A^{01,\alpha} + \underline{\psi}^{[k]} \otimes \underline{\psi}^{[k]} A^{00,\alpha} \right) dt \tag{C.8}$$

$$\underline{\boldsymbol{f}}^{[k],\alpha} = \int_{\breve{I}_k} \left(\dot{\underline{\psi}}^{[k]} D^{1,\alpha}(t) + \underline{\psi}^{[k]} D^{0,\alpha}(t) \right) dt \tag{C.9}$$

and:

$$\underline{\underline{\boldsymbol{Q}}}^{[k],\bar{r}} = \int_{\breve{I}_k} \left(\dot{\underline{\psi}}^{[k]} \otimes \dot{\underline{\psi}}^{[k]} A^{11,\bar{r}} + \dot{\underline{\psi}}^{[k]} \otimes \underline{\psi}^{[k]} A^{10,\bar{r}} + \underline{\psi}^{[k]} \otimes \dot{\underline{\psi}}^{[k]} A^{01,\bar{r}} + \underline{\psi}^{[k]} \otimes \underline{\psi}^{[k]} A^{00,\bar{r}} \right) dt \tag{C.10}$$

$$\underline{\boldsymbol{f}}^{[k],\bar{r}} = \int_{\breve{I}_k} \left(\dot{\underline{\psi}}^{[k]} D^{1,\bar{r}}(t) + \underline{\psi}^{[k]} D^{0,\bar{r}}(t) \right) dt \tag{C.11}$$

By defining the matrices that transmit the continuity between the elements when using the Discontinuous Galerkin approach for the kinematic hardening terms as:

$$\begin{aligned} \underline{\underline{\boldsymbol{\mathcal{L}}}}^{[k],\alpha} &= 1.1 \max(\underline{\underline{\boldsymbol{Q}}}^{[k],\alpha}) \underline{\psi}(t_{k-1}) \otimes \underline{\psi}(t_{k-1}^+) \\ \underline{\underline{\boldsymbol{\mathcal{R}}}}^{[k],\alpha} &= 1.1 \max(\underline{\underline{\boldsymbol{Q}}}^{[k],\alpha}) \underline{\psi}(t_{k-1}) \otimes \underline{\psi}(t_{k-1}^-) \end{aligned} \tag{C.12}$$

and the isotropic hardening ones by:

$$\begin{aligned} \underline{\underline{\boldsymbol{\mathcal{L}}}}^{[k],\bar{r}} &= 1.1 \max(\underline{\underline{\boldsymbol{Q}}}^{[k],\bar{r}}) \underline{\psi}(t_{k-1}) \otimes \underline{\psi}(t_{k-1}^+) \\ \underline{\underline{\boldsymbol{\mathcal{R}}}}^{[k],\bar{r}} &= 1.1 \max(\underline{\underline{\boldsymbol{Q}}}^{[k],\bar{r}}) \underline{\psi}(t_{k-1}) \otimes \underline{\psi}(t_{k-1}^-) \end{aligned} \tag{C.13}$$

we finally obtain the discretized equations in order to determine the temporal functions:

$$\left(\underline{\underline{\boldsymbol{Q}}}^{[k],\alpha} + \underline{\underline{\boldsymbol{\mathcal{L}}}}^{[k],\alpha} \right) \underline{\boldsymbol{\lambda}}^{[k],\alpha} = \underline{\underline{\boldsymbol{\mathcal{R}}}}^{[k],\alpha} \underline{\boldsymbol{\lambda}}^{[k-1],\alpha} + \underline{\boldsymbol{f}}^{[k],\alpha} \tag{C.14}$$

$$\left(\underline{\underline{\boldsymbol{Q}}}^{[k],\bar{r}} + \underline{\underline{\boldsymbol{\mathcal{L}}}}^{[k],\bar{r}} \right) \underline{\boldsymbol{\lambda}}^{[k],\bar{r}} = \underline{\underline{\boldsymbol{\mathcal{R}}}}^{[k],\bar{r}} \underline{\boldsymbol{\lambda}}^{[k-1],\bar{r}} + \underline{\boldsymbol{f}}^{[k],\bar{r}} \tag{C.15}$$

where $\underline{\boldsymbol{\lambda}}^{[k],\alpha}$ and $\underline{\boldsymbol{\lambda}}^{[k],\bar{r}}$ correspond to the nodal values of the time function associated to the kinematic and isotropic hardening at time step k.

Appendix D

Temporal multiscale approximation applied to the preliminary step when dealing with elasto-visco-plasticity

The preliminary step consists in the actualization of the temporal PGD functions of the low-rank approximation of the global stage quantities while maintaining the spatial functions fixed, in order to improve the approximation and accelerate the convergence.

D.1 Temporal functions associated to the admissibility problem

First we consider the actualization of the temporal PGD functions associated to the admissibility quantities, i.e, the stress, strain and plastic deformation. Mathematically this is consists in finding corrective temporal functions $\Delta\lambda_i$ with $i \in [1, ..., m+1]$ such as:

$\forall i \in [1, ..., m+1]$,

$$\lambda_i \leftarrow \lambda_i + \Delta\lambda_i \tag{D.1}$$

As presented in chapter 4, when considering the equation (D.1) the minimization problem (5.70) can be rewritten as:

$$\{\Delta\lambda_i\}_{i=1}^{m+1} = \underset{\{\Delta\lambda_i\}_{i=1}^{m+1} \in \mathcal{U}^T}{\arg\min} \left|\left|\left| \sum_{i=1}^{m+1} \bar{\underline{\varepsilon}}_i^p \Delta\dot{\lambda}_i - \mathbb{H}_\sigma : \sum_{i=1}^{m+1} \bar{\underline{\sigma}}_i \Delta\lambda_i + \bar{\underline{\Delta}}_{n+1} \right|\right|\right|_{\mathbb{H}_\sigma^{-1}}^2 \tag{D.2}$$

In what follows the determination of the macro and micro functions needed for the multiscale approximation of the corrective temporal functions is presented.

• **Macro functions determination**:

As for the case of the enrichment step, the first objective is to determine the macro functions associated to the temporal corrective terms, for this, we consider that $\ell - 1$ sub-mode have already been computed so that the minimization problem can be rewritten as:

$$\left\{\Delta\bar{\lambda}_{i,\ell}\right\}_{i=1}^{m+1} = \underset{\{\Delta\bar{\lambda}_{i,\ell}\}_{i=1}^{m+1} \in \mathcal{U}^T}{\arg\min} \left|\left|\left| \sum_{s=1}^{\ell-1} \sum_{i=1}^{m+1} \bar{\underline{\varepsilon}}_i^p \left(\Delta\dot{\bar{\lambda}}_{i,s}(t) + \Delta\dot{\bar{\lambda}}_{i,\ell}(t)\right) - \mathbb{H}_\sigma : \sum_{s=1}^{\ell-1} \sum_{i=1}^{m+1} \bar{\underline{\sigma}}_i \left(\Delta\bar{\lambda}_{i,s}(t) + \Delta\bar{\lambda}_{i,\ell}(t)\right) + \bar{\underline{\Delta}}_{n+1} \right|\right|\right|_{\mathbb{H}_\sigma^{-1}}^2$$

By rearranging the known and unknown terms, the above expression can be reduced to the main problem

needed to be solved:

$$\left\{\Delta\bar{\lambda}_{i,\ell}\right\}_{i=1}^{m+1} = \underset{\{\Delta\bar{\lambda}_{i,\ell}\}_{i=1}^{m+1}\in\mathcal{U}^T}{\arg\min} \left|\left|\left|\sum_{i=1}^{m+1}\bar{\underline{\underline{\varepsilon}}}_i^p\Delta\dot{\bar{\lambda}}_{i,\ell}(t) - \mathbb{H}_\sigma : \sum_{i=1}^{m+1}\bar{\underline{\underline{\sigma}}}_i\Delta\bar{\lambda}_{i,\ell}(t) + \bar{\underline{\underline{\Delta}}}_{n+1}^{\ell}\right|\right|\right|_{\mathbb{H}_\sigma^{-1}}^2 \tag{D.3}$$

with the residual term given by:

$$\bar{\underline{\underline{\Delta}}}_{n+1}^{\ell} = \underline{\underline{\Delta}}_{n+1} + \sum_{s=1}^{\ell-1}\sum_{i=1}^{m+1}\bar{\underline{\underline{\varepsilon}}}_i^p\Delta\dot{\bar{\lambda}}_{i,s}(t) - \mathbb{H}_\sigma : \bar{\underline{\underline{\sigma}}}_i\Delta\bar{\lambda}_{i,s}(t) \tag{D.4}$$

By defining the following constants and temporal functions as follows:

$\forall(i,j)\in[1,...,m+1]\,,$

$$\begin{aligned} A_{ij}^{11} &= \int_\Omega \bar{\underline{\underline{\varepsilon}}}_i^p : \mathbb{H}_\sigma^{-1} : \bar{\underline{\underline{\varepsilon}}}_j^p \, d\Omega \\ A_{ij}^{10} &= -\int_\Omega \bar{\underline{\underline{\varepsilon}}}_i^p : \bar{\underline{\underline{\sigma}}}_j \, d\Omega \qquad\qquad D_i^1(t) = -\int_\Omega (\bar{\underline{\underline{\varepsilon}}}_i^p : \mathbb{H}_\sigma^{-1} : \underline{\underline{\Delta}}_{n+1}) \, d\Omega \\ A_{ij}^{01} &= -\int_\Omega \bar{\underline{\underline{\sigma}}}_i : \bar{\underline{\underline{\varepsilon}}}_j^p \, d\Omega \qquad , \qquad D_i^0(t) = \int_\Omega \bar{\underline{\underline{\sigma}}}_i : \underline{\underline{\Delta}}_{n+1} \, d\Omega \\ A_{ij}^{00} &= \int_\Omega (\mathbb{H}_\sigma : \bar{\underline{\underline{\sigma}}}_i) : \bar{\underline{\underline{\sigma}}}_j \, d\Omega \end{aligned} \tag{D.5}$$

and developing the minimization problem we obtain:

$\forall\delta\Delta\bar{\lambda}_{i,\ell}\in\mathcal{U}^T,\ \forall i\in[1,...,m+1]\,,$

$$\begin{aligned} \int_I \sum_{i=1}^{m+1}\sum_{j=1}^{m+1} \delta\Delta\dot{\bar{\lambda}}_{i,\ell}\left(\Delta\dot{\bar{\lambda}}_{j,\ell}A_{ij}^{11} + \Delta\bar{\lambda}_{j,\ell}A_{ij}^{10}\right) + \delta\Delta\bar{\lambda}_{i,\ell}\left(\Delta\dot{\bar{\lambda}}_{j,\ell}A_{ij}^{01} + \Delta\bar{\lambda}_{j,\ell}A_{ij}^{00}\right) \, dt = \\ \int_I \sum_{i=1}^{m+1} \delta\Delta\dot{\bar{\lambda}}_{i,\ell}R_{i,\ell}^1(t) + \delta\Delta\bar{\lambda}_{i,\ell}R_{i,\ell}^0(t) \, dt \end{aligned} \tag{D.6}$$

where the residual temporal functions at sub-mode ℓ are given by:

$\forall i\in[1,...,m+1]\,,$

$$\begin{aligned} R_{i,\ell}^1(t) &= D_i^1(t) - \sum_{s=1}^{\ell-1}\sum_{j=1}^{m+1}\left(\Delta\dot{\bar{\lambda}}_{j,s}A_{ij}^{11} + \Delta\bar{\lambda}_{j,s}A_{ij}^{10}\right) \\ R_{i,\ell}^0(t) &= D_i^0(t) - \sum_{s=1}^{\ell-1}\sum_{j=1}^{m+1}\left(\Delta\dot{\bar{\lambda}}_{j,s}A_{ij}^{01} + \Delta\bar{\lambda}_{j,s}A_{ij}^{00}\right) \end{aligned} \tag{D.7}$$

By developing the corresponding equations and introducing the new multiscale approximation of equations (5.33) and (5.34) into (D.6), we obtain the following discretized linear system that allows to determine the value of the macro amplitudes:

$$\underline{\underline{\boldsymbol{Q}}}_{up,\ell}\,\underline{\boldsymbol{A}}_{up,\ell} = \underline{\boldsymbol{f}}_{up,\ell} \tag{D.8}$$

where $\underline{\boldsymbol{A}}_{up,\ell}$ correspond to the nodal values in time of the macro functions associated to all the actualized modes at the sub-mode "ℓ".

$$\underline{\boldsymbol{A}}_{up,\ell} = \begin{bmatrix} \underline{\boldsymbol{A}}_{1,\ell} \\ \underline{\boldsymbol{A}}_{2,\ell} \\ \vdots \\ \\ \underline{\boldsymbol{A}}_{m+1,\ell} \end{bmatrix} \tag{D.9}$$

and the matrices and vectors given by:

$$\underline{\underline{\boldsymbol{Q}}}_{up,\ell} = \begin{bmatrix} \underline{\underline{\boldsymbol{Q}}}_{11,\ell} & \underline{\underline{\boldsymbol{Q}}}_{12,\ell} & \cdots & \underline{\underline{\boldsymbol{Q}}}_{1\ m+1,\ell} \\ \underline{\underline{\boldsymbol{Q}}}_{21,\ell} & \ddots & & \\ \vdots & & & \\ \underline{\underline{\boldsymbol{Q}}}_{m+1\ 1,\ell} & & & \underline{\underline{\boldsymbol{Q}}}_{m+1\ m+1,\ell} \end{bmatrix} \quad , \quad \underline{\boldsymbol{f}}_{up,\ell} = \begin{bmatrix} \underline{\boldsymbol{f}}_{1,\ell} \\ \underline{\boldsymbol{f}}_{2,\ell} \\ \vdots \\ \\ \underline{\boldsymbol{f}}_{m+1,\ell} \end{bmatrix}$$

where the different terms are given by:
$\forall (i,j) \in [1, ..., m+1]$,

$$\begin{aligned} &\underline{\underline{\boldsymbol{Q}}}_{ij,\ell} = \\ &\bigoplus_{k=1}^{N_\ell} \int_{I^M_{k,\ell}} \left[\left(\dot{\underline{n}}_{i,\ell} \dot{\underline{n}}^T_{j,\ell} \otimes \underline{\Psi}^{[k]}_\ell (\underline{\Psi}^{[k]}_\ell)^T \right) + \left(\dot{\underline{n}}_{i,\ell} \underline{n}^T_{j,\ell} \otimes \underline{\Psi}^{[k]}_\ell (\dot{\underline{\Psi}}^{[k]}_\ell)^T \right) + \left(\underline{n}_{i,\ell} \dot{\underline{n}}^T_{j,\ell} \otimes \dot{\underline{\Psi}}^{[k]}_\ell (\underline{\Psi}^{[k]}_\ell)^T \right) + \left(\underline{n}_{i,\ell} \underline{n}^T_{j,\ell} \otimes \dot{\underline{\Psi}}^{[k]}_\ell (\dot{\underline{\Psi}}^{[k]}_\ell)^T \right) \right] A^{11}_{ij} \\ &+ \left[\left(\dot{\underline{n}}_{i,\ell} \underline{n}^T_{j,\ell} \otimes \underline{\Psi}^{[k]}_\ell (\underline{\Psi}^{[k]}_\ell)^T \right) + \left(\underline{n}_{i,\ell} \underline{n}^T_{j,\ell} \otimes \dot{\underline{\Psi}}^{[k]}_\ell (\underline{\Psi}^{[k]}_\ell)^T \right) \right] A^{10}_{ij} + \left[\left(\underline{n}_{i,\ell} \dot{\underline{n}}^T_{j,\ell} \otimes \underline{\Psi}^{[k]}_\ell (\underline{\Psi}^{[k]}_\ell)^T \right) + \left(\underline{n}_{i,\ell} \underline{n}^T_{j,\ell} \otimes \underline{\Psi}^{[k]}_\ell (\dot{\underline{\Psi}}^{[k]}_\ell)^T \right) \right] A^{01}_{ij} \\ &+ \left[\underline{n}_{i,\ell} \underline{n}^T_{j,\ell} \otimes \underline{\Psi}^{[k]}_\ell (\underline{\Psi}^{[k]}_\ell)^T \right] A^{00}_{ij} \, dt \end{aligned} \tag{D.10}$$

$$\underline{\boldsymbol{f}}_{i,\ell} = \bigoplus_{k=1}^{N_\ell} \int_{I^M_{k,\ell}} \left[\left(\underline{\underline{\boldsymbol{I}}} \otimes \underline{\Psi}^{[k]}_\ell \right) \dot{\underline{n}}_{i,\ell} + \left(\underline{\underline{\boldsymbol{I}}} \otimes \dot{\underline{\Psi}}^{[k]}_\ell \right) \underline{n}_{i,\ell} \right] R^1_{i,\ell}(t) + \left(\underline{\underline{\boldsymbol{I}}} \otimes \underline{\Psi}^{[k]}_\ell \right) \underline{n}_{i,\ell} R^0_{i,\ell}(t) \, dt \tag{D.11}$$

Once the macro problem stated, the only variables that remains to be determined are the micro functions. The procedure for their computation is shown below.

• **Micro functions determination**:

In order to obtain the micro functions, we simple reconsider the constitutive relation error related to the admissibility problem given as:

$$J^p = \left| \left| \left| \sum_{i=1}^{m+1} \underline{\underline{\bar{\varepsilon}}}^p_i \Delta \dot{\bar{\lambda}}_{i,\ell}(t) - \mathbb{H}_\sigma : \sum_{i=1}^{m+1} \underline{\underline{\bar{\sigma}}}_i \Delta \bar{\lambda}_{i,\ell}(t) + \underline{\underline{\bar{\Delta}}}^\ell_{n+1} \right| \right|^2_{\mathbb{H}^{-1}_\sigma} \tag{D.12}$$

By developing the above functional we can write:

$$J^p = \int_{\Omega \times I} \left(\sum_{i=1}^{m+1} \underline{\underline{\bar{\varepsilon}}}^p_i \Delta \dot{\bar{\lambda}}_{i,\ell} - \mathbb{H}_\sigma : \sum_{i=1}^{m+1} \underline{\underline{\bar{\sigma}}}_i \Delta \bar{\lambda}_{i,\ell} + \underline{\underline{\bar{\Delta}}}^\ell_{n+1} \right) : \mathbb{H}^{-1}_\sigma : \left(\sum_{j=1}^{m+1} \underline{\underline{\bar{\varepsilon}}}^p_j \Delta \dot{\bar{\lambda}}_{j,\ell} - \mathbb{H}_\sigma : \sum_{j=1}^{m+1} \underline{\underline{\bar{\sigma}}}_j \Delta \bar{\lambda}_{j,\ell} + \underline{\underline{\bar{\Delta}}}^\ell_{n+1} \right) d\Omega dt \tag{D.13}$$

where the above expression can by simplified by using the terms given in (D.5) and (D.7) as follows:

$$\begin{aligned} J^p = &\int_{\Omega \times I} \underline{\underline{\bar{\Delta}}}^\ell_{n+1} : \mathbb{H}^{-1}_\sigma : \underline{\underline{\bar{\Delta}}}^\ell_{n+1} \, d\Omega dt + \int_I \sum_{i=1}^{m+1} \sum_{j=1}^{m+1} \Delta \dot{\bar{\lambda}}_{i,\ell} \left(\Delta \dot{\bar{\lambda}}_{j,\ell} A^{11}_{ij} + \Delta \bar{\lambda}_{j,\ell} A^{10}_{ij} \right) + \Delta \bar{\lambda}_{i,\ell} \left(\Delta \dot{\bar{\lambda}}_{j,\ell} A^{10} + \Delta \bar{\lambda}_{j,\ell} A^{00} \right) \\ &- 2 \sum_{i=1}^{m+1} \left(\Delta \dot{\bar{\lambda}}_{i,\ell} R^1_{i,\ell}(t) + \Delta \bar{\lambda}_{i,\ell} R^0_{i,\ell}(t) \right) \, dt \end{aligned} \tag{D.14}$$

By introducing the multiscale approximation of equations (5.33) and (5.34) into (D.14) and using the amplitude solution of equation (D.8) we have:

$$J^p = \int_I \underline{\underline{\bar{\Delta}}}^{\ell}_{n+1} : \mathbb{H}_{\sigma}^{-1} : \underline{\underline{\bar{\Delta}}}^{\ell}_{n+1} \, d\Omega dt - \underline{\boldsymbol{f}}^T_{up,\ell} \underline{\underline{\boldsymbol{Q}}}^{-1}_{up,\ell} \underline{\boldsymbol{f}}_{up,\ell}$$

which states that the micro functions must be determined such that they maximize $\underline{\boldsymbol{f}}^T_{up,\ell} \underline{\underline{\boldsymbol{Q}}}^{-1}_{up,\ell} \underline{\boldsymbol{f}}_{up,\ell}$. By applying the same approximation as introduced in the enrichment step case (see section 5.3.2.1), to avoid an iterative determination of the micro functions, we define the following functional:

$$g_\ell = \underline{\boldsymbol{f}}^T_{up,\ell} \underline{\boldsymbol{f}}_{up,\ell} \approx \underline{\boldsymbol{f}}^T_{up,\ell} \underline{\underline{\boldsymbol{Q}}}^{-1}_{up,\ell} \underline{\boldsymbol{f}}_{up,\ell}$$

By approximating the above expression applying the same ideas as presented in section 5.3.1 we obtain:

$$g_\ell \approx \sum_{i=1}^{m+1} \sum_{l=1}^{4} \left[\int_I (\dot{h}^{\mathsf{R}}_{i,\ell} \Psi_{l,\ell} + h^{\mathsf{R}}_{i,\ell} \dot{\Psi}_{l,\ell}) R^1_{i,\ell}(t) + h^{\mathsf{R}}_{i,\ell} \Psi_{l,\ell} R^0_{i,\ell}(t) \, dt \right]^2 + \left[\int_I (\dot{h}^{\mathsf{I}}_{i,\ell} \Psi_{l,\ell} + h^{\mathsf{I}}_{i,\ell} \dot{\Psi}_{l,\ell}) R^1_{i,\ell}(t) + h^{\mathsf{I}}_{i,\ell} \Psi_{l,\ell} R^0_{i,\ell}(t) \, dt \right]^2 \tag{D.15}$$

Following the same ideas as for the enrichment step, in order to maximize the above expression we define the micro functions as the superposition of the following terms:

$\forall i \in [1, ..., m+1]$,

$$\begin{aligned} h^{\mathsf{R}}_{i,\ell}(t) &= \sum_{p=1}^{N_{c,\ell}} a^{(\ell)}_{i,p} \int \left[R^1_{i,\ell}(t) \right]_{\mathsf{I},p} dt \qquad & h^{\mathsf{I}}_{i,\ell}(t) &= \sum_{p=1}^{N_{c,\ell}} b^{(\ell)}_{i,p} \int \left[R^1_{i,\ell}(t) \right]_{\mathsf{R},p} dt \\ \dot{h}^{\mathsf{R}}_{i,\ell}(t) &= \sum_{p=1}^{N_{c,\ell}} a^{(\ell)}_{i,p} \left[R^1_{i,\ell}(t) \right]_{\mathsf{I},p} & \dot{h}^{\mathsf{I}}_{i,\ell}(t) &= \sum_{p=1}^{N_{c,\ell}} b^{(\ell)}_{i,p} \left[R^1_{i,\ell}(t) \right]_{\mathsf{R},p} \end{aligned} \tag{D.16}$$

By injecting the expression of the micro functions (D.16) into (D.15) we finally obtain:

$$g_\ell \approx \sum_{i=1}^{m+1} \underline{\boldsymbol{a}}^T_{i,\ell} \underline{\underline{\boldsymbol{N}}}^{\mathsf{R}}_{i,\ell} \underline{\boldsymbol{a}}_{i,\ell} \; + \; \underline{\boldsymbol{b}}^T_{i,\ell} \underline{\underline{\boldsymbol{N}}}^{\mathsf{I}}_{i,\ell} \underline{\boldsymbol{b}}_{i,\ell} \tag{D.17}$$

where the term associated to the symmetric micro function is given by:

$\forall i \in [1, ..., m+1]$,

$$\underbrace{\sum_{l=1}^{4} \left[\sum_{p_1=1}^{N_{c,\ell}} \int_{I_{p_1,\ell}} \left(\left[\Psi_{l,\ell} R^1_{i,\ell}(t) \right]_{p_1} \sum_{p_2=1}^{N_{c,\ell}} a_{i,p_2} \left[R^1_{i,\ell}(t) \right]_{\mathsf{I},p_2} + \left[\dot{\Psi}_{l,\ell} R^1_{i,\ell}(t) + \Psi_{l,\ell} R^0_{i,\ell}(t) \right]_{p_1} \sum_{p_2=1}^{N_{c,\ell}} a_{i,p_2} \int \left[R^1_{i,\ell}(t) \right]_{\mathsf{I},p_2} dt \right) dt \right]^2}_{\underline{\boldsymbol{a}}^T_{i,\ell} \underline{\underline{\boldsymbol{N}}}^{\mathsf{R}}_{i,\ell} \underline{\boldsymbol{a}}_{i,\ell}} \tag{D.18}$$

and the one associated to the antisymmetric micro functions as:

$\forall i \in [1, ..., m+1]$,

$$\underbrace{\sum_{l=1}^{4} \left[\sum_{p_1=1}^{N_{c,\ell}} \int_{I_{p_1,\ell}} \left(\left[\Psi_{l,\ell} R^1_{i,\ell}(t) \right]_{p_1} \sum_{p_2=1}^{N_{c,\ell}} b_{i,p_2} \left[R^1_{i,\ell}(t) \right]_{\mathsf{R},p_2} + \left[\dot{\Psi}_{l,\ell} R^1_{i,\ell}(t) + \Psi_{l,\ell} R^0_{i,\ell}(t) \right]_{p_1} \sum_{p_2=1}^{N_{c,\ell}} b_{i,p_2} \int \left[R^1_{i,\ell}(t) \right]_{\mathsf{R},p_2} dt \right) dt \right]^2}_{\underline{\boldsymbol{b}}^T_{i,\ell} \underline{\underline{\boldsymbol{N}}}^{\mathsf{I}}_{i,\ell} \underline{\boldsymbol{b}}_{i,\ell}} \tag{D.19}$$

where the vectors $\underline{\boldsymbol{a}}_{i,\ell}$ and $\underline{\boldsymbol{b}}_{i,\ell}$ are determined by calculating the eigenvector of matrices $\underline{\underline{\boldsymbol{N}}}^{\mathsf{R}}_{i,\ell}$ and $\underline{\underline{\boldsymbol{N}}}^{\mathsf{I}}_{i,\ell}$ associated to the their maximum eigenvalue for each actualized temporal PGD mode for the construction of the micro functions.

D.2 Internal variables

As presented in chapter 4, the actualization of the temporal functions associated to the internal variables bring out some numerical issues due to bad conditioning of the constructed matrices. To avoid this issue only the last PGD mode is actualized, this is:

$$\begin{aligned}\lambda^{\alpha}_{m+1} &\leftarrow \lambda^{\alpha}_{m+1} + \Delta\lambda^{\alpha}_{m+1}\\ \lambda^{\bar{r}}_{m+1} &\leftarrow \lambda^{\bar{r}}_{m+1} + \Delta\lambda^{\bar{r}}_{m+1}\end{aligned} \tag{D.20}$$

and therefore the minimization problem to be solved is written as:

$$\begin{aligned}\left\{\Delta\lambda^{\alpha}_{m+1}\right\} &= \underset{\Delta\lambda^{\alpha}_{m+1}\in\mathcal{U}^T}{\text{arg min}} \left|\left|\left|\underline{\underline{\bar{\alpha}}}_{m+1}\Delta\dot{\lambda}^{\alpha}_{m+1} + \mathbb{H}_{\beta} : C\underline{\underline{\bar{\alpha}}}_{m+1}\Delta\lambda^{\alpha}_{m+1} + \underline{\underline{\Delta}}^{\alpha}_{n+1}\right|\right|\right|^2_{\mathbb{H}_{\beta}^{-1}}\\ \left\{\Delta\lambda^{\bar{r}}_{m+1}\right\} &= \underset{\Delta\lambda^{\bar{r}}_{m+1}\in\mathcal{U}^T}{\text{arg min}} \left|\left|\left|\bar{\bar{r}}_{m+1}\Delta\dot{\lambda}^{\bar{r}}_{m+1} + H_{\bar{R}}R_{\infty}\bar{\bar{r}}_{m+1}\Delta\lambda^{\bar{r}}_{m+1} + \underline{\underline{\Delta}}^{\bar{r}}_{n+1}\right|\right|\right|^2_{H_{\bar{R}}^{-1}}\end{aligned} \tag{D.21}$$

As can be seen, the above formulation is equivalent to that of the enrichment step, which in turn follows the same procedure outlined for the enrichment step applied to the admissibility quantities as presented in section 5.3.2.1.1, therefore, its presentation herein is omitted.

Bibliography

[Abbasi et al., 2020] Abbasi, M., Iapichino, L., Besselink, B., Schilders, W., and van de Wouw, N. (2020). Error estimation in reduced basis method for systems with time-varying and nonlinear boundary conditions. *Computer Methods in Applied Mechanics and Engineering*, 360:112688.

[Abrahamson and Silva, 2008] Abrahamson, N. and Silva, W. (2008). Summary of the Abrahamson & Silva NGA Ground-Motion Relations. *Earthquake Spectra*, 24(1):67–97.

[Akkar et al., 2014] Akkar, S., Sandikkaya, M. A., Senyurt, M., Azari Sisi, A., Ay, B. O., Traversa, P., Douglas, J., Cotton, F., Luzi, L., Hernandez, B., and Godey, S. (2014). Reference database for seismic ground-motion in europe (resorce). *Bulletin of the Seismological Society of America*, 12:311–339.

[Alart and Dureisseix, 2008] Alart, P. and Dureisseix, D. (2008). A scalable multiscale LATIN method adapted to nonsmooth discrete media. *Computer Methods in Applied Mechanics and Engineering*, 197(5):319–331.

[Allaire, 2005] Allaire, G. (2005). *Analyse numérique et optimisation: Une introduction à la modélisation mathématique et à la simulation numérique*. Editions Ecole Polytechnique.

[Allix, 2013] Allix, O. (2013). The bounded rate concept: A framework to deal with objective failure predictions in dynamic within a local constitutive model. *International Journal of Damage Mechanics*, 22(6):808–828.

[Allix and Deü, 1997] Allix, O. and Deü, J.-F. (1997). Delayed-damage modelling for fracture prediction of laminated composites under dynamic loading. *Engineering Transactions*, 45(1):29–46.

[Allix et al., 2003] Allix, O., Feissel, P., and Thévenet, P. (2003). A delay damage mesomodel of laminates under dynamic loading: basic aspects and identification issues. *Computers & Structures*, 81(12):1177–1191.

[Ammar et al., 2012] Ammar, A., Chinesta, F., Cueto, E., and Doblaré, M. (2012). Proper Generalized Decomposition of time-multiscale models. *International Journal for Numerical Methods in Engineering*, 90(5):569–596.

[Ammar et al., 2010] Ammar, A., Chinesta, F., and Falco, A. (2010). On the convergence of a greedy rank-one update algorithm for a class of linear systems. *Archives of Computational Methods in Engineering*, 17(4):473–486.

[Ammar et al., 2006] Ammar, A., Mokdad, B., Chinesta, F., and Keunings, R. (2006). A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modeling of complex fluids. *Journal of Non-Newtonian Fluid Mechanics*, 139(3):153–176.

[Amsallem and Farhat, 2008] Amsallem, D. and Farhat, C. (2008). Interpolation method for adapting reduced-order models and application to aeroelasticity. *AIAA journal*, 46(7):1803–1813.

[Amsallem and Farhat, 2014] Amsallem, D. and Farhat, C. (2014). On the stability of reduced-order linearized computational fluid dynamics models based on POD and Galerkin projection: descriptor vs non-descriptor forms. In *Reduced order methods for modeling and computational reduction*, pages 215–233. Springer.

[Argyris and Scharpf, 1969] Argyris, J. and Scharpf, D. (1969). Finite elements in time and space. *Nuclear Engineering and Design*, 10(4):456–464.

[Arzt and Ladevèze, 1994] Arzt, M. and Ladevèze, P. (1994). *Approche des phénomenes cycliques par la méthode à grand incrément de temps*. PhD thesis, Cachan, Ecole normale supérieure.

[Baker and Jayaram, 2008] Baker, J. W. and Jayaram, N. (2008). Correlation of spectral acceleration values from nga ground motion models. *Earthquake Spectra*, 24(1):299–317.

[Barbarulo, 2012] Barbarulo, A. (2012). *On a PGD model order reduction technique for mid-frequency acoustic*. PhD thesis.

[Barbarulo et al., 2013] Barbarulo, A., Ladevèze, P., Riou, H., and Graveleau, M. (2013). Proper generalized decomposition and variational theory of complex rays: an alliance to consider uncertainties over mid and high broad frequency bands. In *11e colloque national en calcul des structures*.

[Barbarulo et al., 2014a] Barbarulo, A., Ladevèze, P., Riou, H., and Kovalevsky, L. (2014a). Proper generalized decomposition applied to linear acoustic: a new tool for broad band calculation. *Journal of Sound and Vibration*, 333(11):2422–2431.

[Barbarulo et al., 2014b] Barbarulo, A., Riou, H., Cattabiani, A., Helluy, G., and Ladevèze, P. (2014b). Proper generalized decomposition, an useful tool to consider multi-parametric uncertainties over mid-frequency broad bands.

[Barbarulo et al., 2014c] Barbarulo, A., Riou, H., Kovalevsky, L., and Ladevèze, P. (2014c). PGD-VTCR: a reduced order model technique to solve medium frequency broad band problems on complex acoustical systems. *Strojniški vestnik-Journal of Mechanical Engineering*, 60(5):307–313.

[Barrault et al., 2004] Barrault, M., Maday, Y., Nguyen, N. C., and Patera, A. T. (2004). An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathématique*, 339(9):667–672.

[Bazant et al., 1984] Bazant, Z. P., Belytschko, T. B., Chang, T.-P., et al. (1984). Continuum theory for strain-softening. *Journal of Engineering Mechanics*, 110(12):1666–1692.

[Bellen and Zennaro, 1989] Bellen, A. and Zennaro, M. (1989). Parallel algorithms for initial-value problems for difference and differential equations. *Journal of Computational and applied mathematics*, 25(3):341–350.

[Berkooz et al., 1993] Berkooz, G., Holmes, P., and Lumley, J. L. (1993). The proper orthogonal decomposition in the analysis of turbulent flows. *Annual review of fluid mechanics*, 25(1):539–575.

[Beylkin and Mohlenkamp, 2005] Beylkin, G. and Mohlenkamp, M. J. (2005). Algorithms for numerical analysis in high dimensions. *SIAM Journal on Scientific Computing*, 26(6):2133–2159.

[Bhattacharyya et al., 2019] Bhattacharyya, M., Fau, A., Desmorat, R., Alameddin, S., Néron, D., Ladevèze, P., and Nackenhorst, U. (2019). A kinetic two-scale damage model for high-cycle fatigue simulation using multi-temporal latin framework. *European Journal of Mechanics-A/Solids*, 77:103808.

[Bhattacharyya et al., 2018a] Bhattacharyya, M., Fau, A., Nackenhorst, U., Néron, D., and Ladevèze, P. (2018a). A latin-based model reduction approach for the simulation of cycling damage. *Computational Mechanics*, 62(4):725–743.

[Bhattacharyya et al., 2018b] Bhattacharyya, M., Fau, A., Nackenhorst, U., Néron, D., and Ladevèze, P. (2018b). A model reduction technique in space and time for fatigue simulation. In *Multiscale modeling of heterogeneous structures*, pages 183–203. Springer.

[Bhattacharyya et al., 2018c] Bhattacharyya, M., Fau, A., Nackenhorst, U., Néron, D., and Ladevèze, P. (2018c). A multi-temporal scale model reduction approach for the computation of fatigue damage. *Computer Methods in Applied Mechanics and Engineering*, 340:630–656.

[Billaud-Friess et al., 2014] Billaud-Friess, M., Nouy, A., and Zahm, O. (2014). A tensor approximation method based on ideal minimal residual formulations for the solution of high-dimensional problems. *ESAIM: Mathematical Modelling and Numerical Analysis*, 48(6):1777–1806.

[Bisch et al., 2012] Bisch, P., Carvalho, E., Degee, H., Fajfar, P., Fardis, M. N., Franchin, P., Kreslin, M., Pecker, A., Pinto, P., Plumier, A., Somja, H., and Tsionis, G. (2012). Eurocode 8: Seismic design of buildings – worked examples. In Acun, B., Athanasopoulou, A., Pinto, A., Carvalho, E., and Fardis, M., editors, *Worked examples presented at the Workshop "EC 8: Seismic Design of Buildings", Lisbon, 10-11 Feb. 2011*, number EUR 25204 EN - 2012.

[Boore and Atkinson, 2008] Boore, D. and Atkinson, G. (2008). Ground-motion prediction equations for the average horizontal component of pga, pgv, and 5%-damped psa at spectral periods between 0.01 s and 10.0 s. *Earthquake Spectra*, 24(1):99–138.

[Boucard and Ladevèze, 1999] Boucard, P.-A. and Ladevèze, P. (1999). A multiple solution method for non-linear structural mechanics.

[Boucinha et al., 2014] Boucinha, L., Ammar, A., Gravouil, A., and Nouy, A. (2014). Ideal minimal residual-based proper generalized decomposition for non-symmetric multi-field models–application to transient elastodynamics in space-time domain. *Computer Methods in Applied Mechanics and Engineering*, 273:56–76.

[Boucinha et al., 2013] Boucinha, L., Gravouil, A., and Ammar, A. (2013). Space–time proper generalized decompositions for the resolution of transient elastodynamic models. *Computer Methods in Applied Mechanics and Engineering*, 255:67–88.

[Bourgat et al., 1988] Bourgat, J.-F., Glowinski, R., Le Tallec, P., and Vidrascu, M. (1988). Variational formulation and algorithm for trace operation in domain decomposition calculations.

[Canuto et al., 2009] Canuto, C., Tonn, T., and Urban, K. (2009). A posteriori error analysis of the reduced basis method for nonaffine parametrized nonlinear pdes. *SIAM Journal on Numerical Analysis*, 47(3):2001–2022.

[Capaldo, 2015] Capaldo, M. (2015). *A new approximation framework for PGD-based nonlinear solvers*. PhD thesis.

[Capaldo et al., 2017] Capaldo, M., Guidault, P.-A., Néron, D., and Ladevèze, P. (2017). The reference point method, a hyperreduction technique: Application to pgd-based nonlinear model reduction. *Computer Methods in Applied Mechanics and Engineering*, 322:483–514.

[Cettour-Janet, 2019] Cettour-Janet, R. (2019). *Modelling the vibrational response and acoustic radiation of the railway tracks*. PhD thesis, Paris Saclay.

[Chaboche, 1989] Chaboche, J.-L. (1989). Constitutive equations for cyclic plasticity and cyclic viscoplasticity. *International journal of plasticity*, 5(3):247–302.

[Champaney et al., 1997] Champaney, L., Cognard, J.-Y., Dureisseix, D., and Ladevèze, P. (1997). Large scale applications on parallel computers of a mixed domain decomposition method. *Computational Mechanics*, 19(4):253–263.

[Chapman et al., 2017] Chapman, T., Avery, P., Collins, P., and Farhat, C. (2017). Accelerated mesh sampling for the hyper reduction of nonlinear computational models. *International Journal for Numerical Methods in Engineering*, 109(12):1623–1654.

[Charbonnel, 2018] Charbonnel, P.-E. (2018). Analyse temps-fréquence de signaux sismiques : différentes approches et tentatives de modélisation. Technical report, DEN/DANS/DM2S/SEMT/EMSI/NT/2018-62544/A.

[Chartier and Philippe, 1993] Chartier, P. and Philippe, B. (1993). A parallel shooting technique for solving dissipative ode's. *Computing*, 51(3-4):209–236.

[Chaturantabut and Sorensen, 2009] Chaturantabut, S. and Sorensen, D. C. (2009). Discrete empirical interpolation for nonlinear model reduction. In *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, pages 4316–4321. IEEE.

[Chaturantabut and Sorensen, 2010] Chaturantabut, S. and Sorensen, D. C. (2010). Nonlinear model reduction via discrete empirical interpolation. *SIAM Journal on Scientific Computing*, 32(5):2737–2764.

[Chen et al., 2017] Chen, P., Quarteroni, A., and Rozza, G. (2017). Reduced basis methods for uncertainty quantification. *SIAM/ASA Journal on Uncertainty Quantification*, 5(1):813–869.

[Chevreuil and Nouy, 2012] Chevreuil, M. and Nouy, A. (2012). Model order reduction based on proper generalized decomposition for the propagation of uncertainties in structural dynamics. *International Journal for Numerical Methods in Engineering*, 89(2):241–268.

[Chinesta et al., 2010] Chinesta, F., Ammar, A., and Cueto, E. (2010). Recent advances and new challenges in the use of the proper generalized decomposition for solving multidimensional models. *Archives of Computational methods in Engineering*, 17(4):327–350.

[Chinesta et al., 2011a] Chinesta, F., Ammar, A., Leygue, A., and Keunings, R. (2011a). An overview of the proper generalized decomposition with applications in computational rheology. *Journal of Non-Newtonian Fluid Mechanics*, 166(11):578–592.

[Chinesta et al., 2013] Chinesta, F., Keunings, R., and Leygue, A. (2013). *The proper generalized decomposition for advanced numerical simulations: a primer*. Springer Science & Business Media.

[Chinesta and Ladevèze, 2014] Chinesta, F. and Ladevèze, P. (2014). Separated representations and PGD-based model reduction. *Fundamentals and Applications, International Centre for Mechanical Siences, Courses and Lectures*, 554:24.

[Chinesta et al., 2011b] Chinesta, F., Ladevèze, P., and Cueto, E. (2011b). A short review on model order reduction based on proper generalized decomposition. *Archives of Computational Methods in Engineering*, 18(4):395–404.

[Chiou and Youngs, 2008] Chiou, B. and Youngs, R. (2008). An NGA model for the average horizontal component of peak ground motion and response spectra. *Earthquake Spectra*, 24(1):173–215.

[Christensen et al., 1999] Christensen, E. A., Brøns, M., and Sørensen, J. N. (1999). Evaluation of proper orthogonal decomposition–based decomposition techniques applied to parameter-dependent nonturbulent flows. *SIAM Journal on Scientific Computing*, 21(4):1419–1434.

[Clough and Penzien, 2003] Clough, R. W. and Penzien, J. (2003). Dynamics of structures. Berkeley. *CA: Computers and Structures*.

[Cognard and Ladevèze, 1993] Cognard, J.-Y. and Ladevèze, P. (1993). A large time increment approach for cyclic viscoplasticity. *International Journal of plasticity*, 9(2):141–157.

[Cremonesi et al., 2013] Cremonesi, M., Néron, D., Guidault, P.-A., and Ladevèze, P. (2013). A PGD-based homogenization technique for the resolution of nonlinear multiscale problems. *Computer Methods in Applied Mechanics and Engineering*, 267:275–292.

[Deparis and Rozza, 2009] Deparis, S. and Rozza, G. (2009). Reduced basis method for multi-parameter-dependent steady navier–stokes equations: applications to natural convection in a cavity. *Journal of Computational Physics*, 228(12):4359–4378.

[Dokainish and Subbaraj, 1989] Dokainish, M. and Subbaraj, K. (1989). A survey of direct time-integration methods in computational structural dynamics—I. Explicit methods. *Computers & Structures*, 32(6):1371–1386.

[Douglas and Aochi, 2008] Douglas, J. and Aochi, H. (2008). A survey of techniques for predicting earthquake ground motions for engineering purposes. *Surveys in Geophysics*, 29:187–220.

[Dureisseix, 1997] Dureisseix, D. (1997). *Une approche multi-échelles pour des calculs de structures sur ordinateurs à architecture parallèle*. PhD thesis.

[Dureisseix and Ladevèze, 1998] Dureisseix, D. and Ladevèze, P. (1998). A 2-level and mixed domain decomposition approach for structural analysis. *Contemporary mathematics*, 218(1):238–245.

[Dureisseix et al., 2003] Dureisseix, D., Ladevèze, P., and Schrefler, B. A. (2003). A LATIN computational strategy for multiphysics problems: application to poroelasticity. *International Journal for Numerical Methods in Engineering*, 56(10):1489–1510.

[Eftang et al., 2010] Eftang, J. L., Grepl, M. A., and Patera, A. T. (2010). A posteriori error bounds for the empirical interpolation method. *Comptes Rendus Mathematique*, 348(9-10):575–579.

[Elghazouli, 2009] Elghazouli, A. Y., editor (2009). *Seismic Design of Buildings to EUROCODE 8*. Taylor and Francis.

[EUROCODE-8, 2004] EUROCODE-8 (2004). Eurocode 8: Design of structures for earthquake resistance – part 1 : General rules, seismic actions and rules for buildings. Technical Report EN 1998-1 :2004 (E), European Committee For Standardization, Management Centre: rue de Stassart, 36 B-1050 Brussels.

[Everson and Sirovich, 1995] Everson, R. and Sirovich, L. (1995). Karhunen–Loeve procedure for gappy data. *JOSA A*, 12(8):1657–1664.

[Farhat et al., 2014] Farhat, C., Avery, P., Chapman, T., and Cortial, J. (2014). Dimensional reduction of nonlinear finite element dynamic models with finite rotations and energy-based mesh sampling and weighting for computational efficiency. *International Journal for Numerical Methods in Engineering*, 98(9):625–662.

[Farhat et al., 2015] Farhat, C., Chapman, T., and Avery, P. (2015). Structure-preserving, stability, and accuracy properties of the energy-conserving sampling and weighting method for the hyper reduction of nonlinear finite element dynamic models. *International journal for numerical methods in engineering*, 102(5):1077–1110.

[Farhat et al., 2020] Farhat, C., Grimberg, S., Manzoni, A., and Quarteroni, A. (2020). 5 computational bottlenecks for proms: precomputation and hyperreduction. In *Snapshot-Based Methods and Algorithms*, pages 181–244. De Gruyter.

[Farhat et al., 2001] Farhat, C., Lesoinne, M., LeTallec, P., Pierson, K., and Rixen, D. (2001). FETI-DP: a dual–primal unified FETI method—part I: A faster alternative to the two-level FETI method. *International journal for numerical methods in engineering*, 50(7):1523–1544.

[Farhat and Roux, 1992] Farhat, C. and Roux, F.-X. (1992). An unconventional domain decomposition method for an efficient parallel solution of large-scale finite element systems. *SIAM Journal on Scientific and Statistical Computing*, 13(1):379–396.

[Favoretto et al., 2019] Favoretto, B., de Hillerin, C., Bettinotti, O., Oancea, V., and Barbarulo, A. (2019). Reduced order modeling via pgd for highly transient thermal evolutions in additive manufacturing. *Computer Methods in Applied Mechanics and Engineering*, 349:405–430.

[Feeny and Kappagantu, 1998] Feeny, B. and Kappagantu, R. (1998). On the physical interpretation of proper orthogonal modes in vibrations. *Journal of Sound and Vibration*, 211(4):607–616.

[Foerster et al., 2020] Foerster, E., Raimond, E., and Guigueno, Y. (2020). Probabilistic safety assessment for internal and external events/european projects h2020-narsis and fp7-asampsa_e. *EPJ Nuclear Sci. Technol.*, 6:38.

[Galvis and Kang, 2014] Galvis, J. and Kang, S. K. (2014). Spectral multiscale finite element for nonlinear flows in highly heterogeneous media: A reduced basis approach. *Journal of Computational and Applied Mathematics*, 260:494–508.

[Gatti et al., 2018] Gatti, F., Touhami, S., Lopez-Caballero, F., Paolucci, R., Clouteau, D., Fernandes, V. A., Kham, M., and Voldoire, F. (2018). Broad-band 3-d earthquake simulation at nuclear site by an all-embracing source-to-structure approach. *Soil Dynamics and Earthquake Engineering*, 115:263 – 280.

[Gavin, 2001] Gavin, H. (2001). Numerical integration for structural dynamics. *Department of Civil and Environmental Engineering, Duke University: Durham, NC, USA*.

[Glüsmann and Kreuzer, 2009] Glüsmann, P. and Kreuzer, E. (2009). On the application of Karhunen–Loeve transform to transient dynamic systems. *Journal of Sound and Vibration*, 328(4-5):507–519.

[Goodfellow et al., 2016] Goodfellow, I., Bengio, Y., Courville, A., and Bengio, Y. (2016). *Deep learning*, volume 1. MIT press Cambridge.

[Grepl, 2005] Grepl, M. A. (2005). *Reduced-basis approximation a posteriori error estimation for parabolic partial differential equations*. PhD thesis, Massachusetts Institute of Technology.

[Grepl et al., 2007] Grepl, M. A., Maday, Y., Nguyen, N. C., and Patera, A. T. (2007). Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 41(3):575–605.

[Grimberg et al., 2021] Grimberg, S., Farhat, C., Tezaur, R., and Bou-Mosleh, C. (2021). Mesh sampling and weighting for the hyperreduction of nonlinear Petrov–Galerkin reduced-order models with local reduced-order bases. *International Journal for Numerical Methods in Engineering*, 122(7):1846–1874.

[Gupta, 2005] Gupta, J. S. (2005). *Mésodynamique et rupture des composites 3D C/C sous choc: une stratégie numérique dédiée*. PhD thesis, École normale supérieure de Cachan-ENS Cachan.

[Hain et al., 2019] Hain, S., Ohlberger, M., Radic, M., and Urban, K. (2019). A hierarchical a posteriori error estimator for the reduced basis method. *Advances in Computational Mathematics*, 45(5):2191–2214.

[Hesthaven, 2016] Hesthaven, Jan S., R. G. S. B. (2016). *Certified Reduced Basis Methods for Parametrized Partial Differential Equations*. Springer Computational Science & Engineering.

[Heyberger et al., 2012] Heyberger, C., Boucard, P.-A., and Néron, D. (2012). Multiparametric analysis within the proper generalized decomposition framework. *Computational Mechanics*, 49(3):277–289.

[Heyberger et al., 2013] Heyberger, C., Boucard, P.-A., and Néron, D. (2013). A rational strategy for the resolution of parametrized problems in the PGD framework. *Computer Methods in Applied Mechanics and Engineering*, 259:40–49.

[Huerta et al., 2018] Huerta, A., Nadal, E., and Chinesta, F. (2018). Proper generalized decomposition solutions within a domain decomposition strategy. *International Journal for Numerical Methods in Engineering*, 113(13):1972–1994.

[Hulbert and Hughes, 1990] Hulbert, G. M. and Hughes, T. J. (1990). Space-time finite element methods for second-order hyperbolic equations. *Computer Methods in Applied Mechanics and Engineering*, 84(3):327–348.

[Huynh et al., 2010] Huynh, D., Knezevic, D., Chen, Y., Hesthaven, J. S., and Patera, A. (2010). A natural-norm successive constraint method for inf-sup lower bounds. *Computer Methods in Applied Mechanics and Engineering*, 199(29-32):1963–1975.

[Huynh et al., 2007] Huynh, D. B. P., Rozza, G., Sen, S., and Patera, A. T. (2007). A successive constraint linear optimization method for lower bounds of parametric coercivity and inf–sup stability constants. *Comptes Rendus Mathematique*, 345(8):473–478.

[IAEA, 2003] IAEA (2003). *Seismic Design and Qualification for Nuclear Power Plants*. Number NS-G-1.6 in Specific Safety Guides. International Atomic Energy Agency, Vienna.

[Jung et al., 2009] Jung, N., Haasdonk, B., and Kroner, D. (2009). Reduced basis method for quadratically nonlinear transport equations. *International Journal of Computing Science and Mathematics*, 2(4):334–353.

[Karhunen, 1946] Karhunen, K. (1946). Zur spektraltheorie stochastischer prozesse. *Ann. Acad. Sci. Fennicae, AI*, 34.

[Kerfriden, 2008] Kerfriden, P. (2008). *Stratégie de décomposition de domaine à trois échelles pour la simulation du délaminage dans les stratifiés*. PhD thesis.

[Kerfriden et al., 2009] Kerfriden, P., Allix, O., and Gosselet, P. (2009). A three-scale domain decomposition method for the 3d analysis of debonding in laminates. *Computational Mechanics*, 44(3):343–362.

[Kerschen et al., 2005] Kerschen, G., Golinval, J.-c., Vakakis, A. F., and Bergman, L. A. (2005). The method of proper orthogonal decomposition for dynamical characterization and order reduction of mechanical systems: an overview. *Nonlinear Dynamics*, 41(1-3):147–169.

[Kosambi, 1943] Kosambi, D. (1943). Statistics in function space. *Journal of Indian Mathematical Society*, 7:76–88.

[Kunisch and Volkwein, 2001] Kunisch, K. and Volkwein, S. (2001). Galerkin proper orthogonal decomposition methods for parabolic problems. *Numerische Mathematik*, 90(1):117–148.

[Kunisch and Volkwein, 2002] Kunisch, K. and Volkwein, S. (2002). Galerkin proper orthogonal decomposition methods for a general equation in fluid dynamics. *SIAM Journal on Numerical analysis*, 40(2):492–515.

[Kutta, 1901] Kutta, W. (1901). Beitrag zur naherungsweisen integration totaler differentialgleichungen. *Z. Math. Phys.*, 46:435–453.

[Ladevèze, 1985] Ladevèze, P. (1985). Sur une famille d'algorithmes en mécanique des structures. *Comptes-rendus des séances de l'Académie des sciences. Série 2, Mécanique-physique, chimie, sciences de l'univers, sciences de la terre*, 300(2):41–44.

[Ladevèze, 1989] Ladevèze, P. (1989). La méthode à grand incrément de temps pour l'analyse de structures à comportement non linéaire décrit par variables internes. *Comptes rendus de l'Académie des sciences. Série 2, Mécanique, Physique, Chimie, Sciences de l'univers, Sciences de la Terre*, 309(11):1095–1099.

[Ladevèze, 1996] Ladevèze, P. (1996). *Mécanique non linéaire des structures: nouvelle approche et méthodes de calcul non incrémentales*. Hermes.

[Ladevèze, 1999] Ladevèze, P. (1999). *Nonlinear computational structural mechanics: new approaches and non-incremental methods of calculation*. Springer Science & Business Media.

[Ladevèze, 2018] Ladevèze, P. (2018). Sur le traitement de chargements sismiques ou de fatigue en mécanique non linéaire des structures. Technical report, LMT-Cachan (ENS Paris-Saclay/CNRS/Université Paris-Saclay).

[Ladevèze and Dureisseix, 2000] Ladevèze, P. and Dureisseix, D. (2000). A micro/macro approach for parallel computing of heterogeneous structures. *International Journal for Computational Civil and Structural Engineering*, 1:18–28.

[Ladevèze et al., 2001] Ladevèze, P., Loiseau, O., and Dureisseix, D. (2001). A micro–macro and parallel computational strategy for highly heterogeneous structures. *International Journal for Numerical Methods in Engineering*, 52(1-2):121–138.

[Ladevèze and Lorong, 1991] Ladevèze, P. and Lorong, P. (1991). A large time increment approach with domain decomposition for mechanical non linear problem. In *Proceedings of the 10th international conference on computing methods in applied sciences and engineering*, pages 569–578.

[Ladevèze and Moës, 1998] Ladevèze, P. and Moës, N. (1998). A posteriori constitutive relation error estimators for nonlinear finite element analysis and adaptive control. *Studies in Applied Mechanics*, 47:231–256.

[Ladevèze et al., 2007] Ladevèze, P., Néron, D., and Gosselet, P. (2007). On a mixed and multiscale domain decomposition method. *Computer Methods in Applied Mechanics and Engineering*, 196(8):1526–1540.

[Ladevèze et al., 2010a] Ladevèze, P., Néron, D., and Passieux, J.-C. (2010a). On multiscale computational mechanics with time-space homogenization. *Multiscale methods—Bridging the scales in Science and Engineering*, pages 247–282.

[Ladevèze and Nouy, 2002] Ladevèze, P. and Nouy, A. (2002). Une stratégie de calcul multiéchelle avec homogénéisation en espace et en temps. *Comptes Rendus Mécanique*, 330(10):683–689.

[Ladevèze and Nouy, 2003] Ladevèze, P. and Nouy, A. (2003). On a multiscale computational strategy with time and space homogenization for structural mechanics. *Computer Methods in Applied Mechanics and Engineering*, 192(28-30):3061–3087.

[Ladevèze et al., 2018] Ladevèze, P., Paillet, C., and Néron, D. (2018). Extended-PGD model reduction for nonlinear solid mechanics problems involving many parameters. In *Advances in Computational Plasticity*, pages 201–220. Springer.

[Ladevèze et al., 2010b] Ladevèze, P., Passieux, J.-C., and Néron, D. (2010b). The LATIN multiscale computational method and the proper generalized decomposition. *Computer Methods in Applied Mechanics and Engineering*, 199(21-22):1287–1296.

[Lancieri et al., 2018] Lancieri, M., Bazzuro, P., and Scotti, O. (2018). Spectral matching in time domain: A seismological and engineering analysis. *Bulletin of the Seismological Society of America*, 108(4):1972–1994.

[Lancieri et al., 2012] Lancieri, M., Madariaga, R., and Bonilla, F. (2012). Spectral scaling of the aftershocks of the Tocopilla 2007 earthquake in northern Chile. *Geophysical Journal International*.

[Lasry and Belytschko, 1988] Lasry, D. and Belytschko, T. (1988). Localization limiters in transient problems. *International Journal of Solids and Structures*, 24(6):581–597.

[Le Tallec, 1994] Le Tallec, P. (1994). Domain decomposition methods in computational mechanics. *Computational Mechanics Advances*, 1(2):121–220.

[Lee and Verleysen, 2007] Lee, J. A. and Verleysen, M. (2007). *Nonlinear dimensionality reduction*. Springer Science & Business Media.

[Lemaitre and Chaboche, 1985] Lemaitre, J. and Chaboche, J.-L. (1985). Mécanique des matériaux solides. dunod. *Mechanics of Solid Materials, Springer*.

[Lemaitre and Chaboche, 1994] Lemaitre, J. and Chaboche, J.-L. (1994). *Mechanics of solid materials*. Cambridge university press.

[Lemaitre and Desmorat, 2005] Lemaitre, J. and Desmorat, R. (2005). *Engineering damage mechanics: ductile, creep, fatigue and brittle failures*. Springer Science & Business Media.

[Liang et al., 2002] Liang, Y., Lee, H., Lim, S., Lin, W., Lee, K., and Wu, C. (2002). Proper orthogonal decomposition and its applications—part I: Theory. *Journal of Sound and Vibration*, 252(3):527–544.

[Lieu and Farhat, 2005] Lieu, T. and Farhat, C. (2005). Adaptation of POD-based aeroelastic ROMs for varying mach number and angle of attack: application to a complete F-16 configuration. In *2005 US Air Force T&E Days*, page 7666.

[Lieu and Farhat, 2007] Lieu, T. and Farhat, C. (2007). Adaptation of aeroelastic reduced-order models and application to an F-16 configuration. *AIAA journal*, 45(6):1244–1257.

[Lieu et al., 2005] Lieu, T., Farhat, C., and Lesoinne, M. (2005). POD-based aeroelastic analysis of a complete F-16 configuration: ROM adaptation and demonstration. In *46th AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics and materials conference*, page 2295.

[Lieu and Lesoinne, 2004] Lieu, T. and Lesoinne, M. (2004). Parameter adaptation of reduced order models for three-dimensional flutter analysis. In *42nd AIAA Aerospace Sciences Meeting and Exhibit*, page 888.

[Lions, 2001] Lions, J.-L. (2001). Résolution d'EDP par un schéma en temps "pararéel". *CRASM*, 332(7):661–668.

[Lions, 1988] Lions, P.-L. (1988). On the Schwarz alternating method. I. In *First international symposium on domain decomposition methods for partial differential equations*, volume 1, page 42. Paris, France.

[Lions, 1990] Lions, P.-L. (1990). On the Schwarz alternating method. III: a variant for nonoverlapping subdomains. In *Third international symposium on domain decomposition methods for partial differential equations*, volume 6, pages 202–223. SIAM Philadelphia, PA.

[Loeve, 1948] Loeve, M. (1948). Fonctions aléatoires du second ordre. *Lévy, Processus Stochastiques et Mouvement Brownien, Gauthier-Villars, Paris*.

[Lülf et al., 2015] Lülf, F. A., Tran, D.-M., Matthies, H. G., and Ohayon, R. (2015). An integrated method for the transient solution of reduced order models of geometrically nonlinear structures. *Computational Mechanics*, 55(2):327–344.

[Maday, 2006] Maday, Y. (2006). Reduced basis method for the rapid and reliable solution of partial differential equations.

[Maday et al., 2009] Maday, Y., Nguyen, N. C., Patera, A. T., and Pau, S. (2009). A general multipurpose interpolation procedure: the magic points. *Communications on Pure & Applied Analysis*, 8(1):383.

[Maday et al., 2002] Maday, Y., Patera, A. T., and Turinici, G. (2002). A priori convergence theory for reduced-basis approximations of single-parameter elliptic partial differential equations. *Journal of Scientific Computing*, 17(1-4):437–446.

[Maday and Ronquist, 2004] Maday, Y. and Ronquist, E. M. (2004). The reduced basis element method: application to a thermal fin problem. *SIAM Journal on Scientific Computing*, 26(1):240–258.

[Mandel, 1993] Mandel, J. (1993). Balancing domain decomposition. *Communications in numerical methods in engineering*, 9(3):233–241.

[Marquis, 1989] Marquis, D. (1989). *Phénoménologie et thermodynamique: couplages entre thermoélasticité, plasticité, vieillissement et endommagement*. PhD thesis, Paris 6.

[Mazars, 1984] Mazars, J. (1984). *Application de la mécanique de l'endommagement au comportement non linéaire et à la rupture du béton de structure*. PhD thesis.

[Mazars et al., 1990] Mazars, J., Berthaud, Y., and Ramtani, S. (1990). The unilateral behaviour of damaged concrete. *Engineering Fracture Mechanics*, 35(4-5):629–635.

[Nachar, 2019] Nachar, S. (2019). *Optimisation de structures viscoplastiques par couplage entre métamodèle multi-fidélité et modèles réduits*. PhD thesis, Université Paris-Saclay.

[NEHRP, 2010] NEHRP (2010). Earthquake-resistant design concepts an introduction to the nehrp recommended seismic provisions for new buildings and other structures. Technical Report FEMA P-749, National Institute of Building Sciences, Building Seismic Safety Council Washington, DC. Prepared for the Federal Emergency Management Agency of the U. S. Department of Homeland Security By the National Institute of Building Sciences Building Seismic Safety Council.

[Néron, 2004] Néron, D. (2004). *Sur une stratégie de calcul pour les problèmes multiphysiques*. PhD thesis, École normale supérieure de Cachan-ENS Cachan.

[Néron et al., 2015] Néron, D., Boucard, P.-A., and Relun, N. (2015). Time-space PGD for the rapid solution of 3D nonlinear parametrized problems in the many-query context. *International Journal for Numerical Methods in Engineering*, 103(4):275–292.

[Néron and Dureisseix, 2008] Néron, D. and Dureisseix, D. (2008). A computational strategy for poroelastic problems with a time interface between coupled physics. *International Journal for Numerical Methods in Engineering*, 73(6):783–804.

[Newmark, 1959] Newmark, N. M. (1959). A method of computation for structural dynamics. *Journal of the Engineering Mechanics Division*, 85(3):67–94.

[Nguyen, 2008] Nguyen, N. C. (2008). A multiscale reduced-basis method for parametrized elliptic partial differential equations with multiple scales. *Journal of Computational Physics*, 227(23):9807–9822.

[Nievergelt, 1964] Nievergelt, J. (1964). Parallel methods for integrating ordinary differential equations. *Communications of the ACM*, 7(12):731–733.

[Nikishkov, 2010] Nikishkov, G. P. (2010). *Programming finite elements in JavaTM*. Springer Science & Business Media.

[Niroomandi et al., 2010] Niroomandi, S., Alfaro, I., Cueto, E., and Chinesta, F. (2010). Model order reduction for hyperelastic materials. *International Journal for Numerical Methods in Engineering*, 81(9):1180–1206.

[Nonino et al., 2021] Nonino, M., Ballarin, F., and Rozza, G. (2021). A monolithic and a partitioned reduced basis method for fluid-structure interaction problems. *arXiv preprint arXiv:2104.09882*.

[Nouailletas et al., 2015] Nouailletas, O., Borderie, C. L., Perlot, C., Rivard, P., and Ballivy, G. (2015). Experimental study of crack closure on heterogeneous quasi-brittle material. *Journal of Engineering Mechanics*, 141(11):04015041.

[Nouy, 2003] Nouy, A. (2003). *Une stratégie de calcul multiéchelle avec homogénéisation en temps et en espace pour le calcul de structures fortement hétérogènes*. PhD thesis, École normale supérieure de Cachan-ENS Cachan.

[Nouy, 2008] Nouy, A. (2008). Generalized spectral decomposition method for solving stochastic finite element equations: invariant subspace problem and dedicated algorithms. *Computer Methods in Applied Mechanics and Engineering*, 197(51-52):4718–4736.

[Nouy, 2009] Nouy, A. (2009). Recent developments in spectral stochastic methods for the numerical solution of stochastic partial differential equations. *Archives of Computational Methods in Engineering*, 16(3):251–285.

[Nouy, 2010a] Nouy, A. (2010a). A priori model reduction through proper generalized decomposition for solving time-dependent partial differential equations. *Computer Methods in Applied Mechanics and Engineering*, 199(23-24):1603–1626.

[Nouy, 2010b] Nouy, A. (2010b). Proper generalized decompositions and separated representations for the numerical solution of high dimensional stochastic problems. *Archives of Computational Methods in Engineering*, 17(4):403–434.

[Nouy and Ladevèze, 2004] Nouy, A. and Ladevèze, P. (2004). Multiscale computational strategy with time and space homogenization: a radial-type approximation technique for solving microproblems. *International Journal for Multiscale Computational Engineering*, 2(4).

[Obukhov, 1954] Obukhov, M. A. (1954). Statistical description of continuous fields. *Transactions of the Geophysical International Academy Nauk USSR*, 24:3–42.

[Oumaziz et al., 2018] Oumaziz, P., Gosselet, P., Boucard, P.-A., and Abbas, M. (2018). A parallel non-invasive multiscale strategy for a mixed domain decomposition method with frictional contact. *International Journal for Numerical Methods in Engineering*, 115(8):893–912.

[Oumaziz et al., 2017] Oumaziz, P., Gosselet, P., Boucard, P.-A., and Guinard, S. (2017). A non-invasive implementation of a mixed domain decomposition method for frictional contact problems. *Computational Mechanics*, 60(5):797–812.

[Oumaziz et al., 2021] Oumaziz, P., Gosselet, P., Saavedra, K., and Tardieu, N. (2021). Analysis, improvement and limits of the multiscale LATIN method. *Computer Methods in Applied Mechanics and Engineering*, 384:113955.

[Paillet et al., 2017] Paillet, C., Ladevèze, P., and Néron, D. (2017). Extensions de la méthode PGD pour les problèmes à grand nombre de paramètres. In *13e colloque national en calcul des structures*.

[Paillet et al., 2018a] Paillet, C., Ladevèze, P., and Néron, D. (2018a). Parameter-multiscale pgd methods for high dimensional parametric spaces. In *6th European Conference on Computational Mechanics (ECCM 6)*.

[Paillet et al., 2018b] Paillet, C., Néron, D., and Ladevèze, P. (2018b). A door to model reduction in high-dimensional parameter space. *Comptes Rendus Mécanique*, 346(7):524–531.

[Passieux, 2008] Passieux, J.-C. (2008). *Approximation radiale et méthode LATIN multiéchelle en temps et en espace*. PhD thesis.

[Passieux et al., 2008] Passieux, J.-C., Ladevèze, P., and Néron, D. (2008). Parallel computing features of a multiscale strategy with space and time homogenization. In *Proc. of the 6th International Conference on Engineering Computational Technology*.

[Passieux et al., 2010] Passieux, J.-C., Ladevèze, P., and Néron, D. (2010). A scalable time–space multiscale domain decomposition method: adaptive time scale separation. *Computational Mechanics*, 46(4):621–633.

[Passieux and Périé, 2012] Passieux, J.-C. and Périé, J.-N. (2012). High resolution digital image correlation using proper generalized decomposition: Pgd-dic. *International Journal for Numerical Methods in Engineering*, 92(6):531–550.

[Pearson, 1901] Pearson, K. (1901). Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572.

[Peterson, 1989] Peterson, J. S. (1989). The reduced basis method for incompressible viscous flow calculations. *SIAM Journal on Scientific and Statistical Computing*, 10(4):777–786.

[Power et al., 2008] Power, M., Chiou, B., Abrahamson, N., Bozorgnia, Y., Shantz, T., and Roblee, C. (2008). An overview of the NGA project. *Earthquake Spectra*, 24(1):3–21.

[Prud'Homme et al., 2002] Prud'Homme, C., Rovas, D. V., Veroy, K., Machiels, L., Maday, Y., Patera, A. T., and Turinici, G. (2002). Reliable real-time solution of parametrized partial differential equations: Reduced-basis output bound methods. *J. Fluids Eng.*, 124(1):70–80.

[Prud'Homme et al., 2002] Prud'Homme, C., Rovas, D. V., Veroy, K., and Patera, A. T. (2002). A mathematical and computational framework for reliable real-time solution of parametrized partial differential equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 36(5):747–771.

[Quarteroni and Rozza, 2007] Quarteroni, A. and Rozza, G. (2007). Numerical solution of parametrized Navier–Stokes equations by reduced basis methods. *Numerical Methods for Partial Differential Equations: An International Journal*, 23(4):923–948.

[Quarteroni et al., 2011] Quarteroni, A., Rozza, G., and Manzoni, A. (2011). Certified reduced basis approximation for parametrized partial differential equations and applications. *Journal of Mathematics in Industry*, 1(1):3.

[Relun et al., 2011] Relun, N., Néron, D., and Boucard, P.-A. (2011). Multiscale elastic-viscoplastic computational analysis: a detailed example. *European Journal of Computational Mechanics/Revue Européenne de Mécanique Numérique*, 20(7-8):379–409.

[Relun et al., 2013] Relun, N., Néron, D., and Boucard, P.-A. (2013). A model reduction technique based on the PGD for elastic-viscoplastic computational analysis. *Computational Mechanics*, 51(1):83–92.

[Rezaeian and Der Kiureghian, 2008] Rezaeian, S. and Der Kiureghian, A. (2008). A stochastic ground motion model with separable temporal and spectral nonstationarities. *Earthquake Engineering and Structural Dynamics*, 37:1565–1584.

[Rezaeian and Der Kiureghian, 2010] Rezaeian, S. and Der Kiureghian, A. (2010). Simulation of synthetic ground motions for specified earthquake and site characteristics. *Earthquake Engineering and Structural Dynamics*, 39:1155–1180.

[Richard and Ragueneau, 2013] Richard, B. and Ragueneau, F. (2013). Continuum damage mechanics based model for quasi brittle materials subjected to cyclic loadings: Formulation, numerical implementation and applications. *Engineering Fracture Mechanics*, 98:383–406.

[Richard et al., 2010] Richard, B., Ragueneau, F., Cremona, C., and Adelaide, L. (2010). Isotropic continuum damage mechanics for concrete under cyclic loading: stiffness recovery, inelastic strains and frictional sliding. *Engineering Fracture Mechanics*, 77(8):1203–1223.

[Riou et al., 2013] Riou, H., Barbarulo, A., Kovalevsky, L., and Ladevèze, P. (2013). PGD-TVRC: une méthode de réduction de modèle pour les vibrations moyennes fréquences. In *11e colloque national en calcul des structures*.

[Rossetto et al., 2016] Rossetto, T., Gehl, P., Minas, S., Galasso, C., Duffour, P., Douglas, J., and Cook, O. (2016). Fracas: A capacity spectrum approach for seismic fragility assessment including record-to-record variability. *Engineering Structures*, 125:337 – 348.

[Rozza, 2011] Rozza, G. (2011). Reduced basis approximation and error bounds for potential flows in parametrized geometries. *Communications in Computational Physics*, 9(1):1–48.

[Rozza et al., 2007] Rozza, G., Huynh, D. B. P., and Patera, A. T. (2007). Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations. *Archives of Computational Methods in Engineering*, 15(3):1.

[Rozza et al., 2009] Rozza, G., Nguyen, C., Patera, A. T., and Deparis, S. (2009). Reduced basis methods and a posteriori error estimators for heat transfer problems. In *Heat Transfer Summer Conference*, volume 43574, pages 753–762.

[Runge, 1895] Runge, C. (1895). Über die numerische auflösung von differentialgleichungen. *Mathematische Annalen*, 46(2):167–178.

[Ryckelynck, 2005] Ryckelynck, D. (2005). A priori hyperreduction method: an adaptive approach. *Journal of Computational Physics*, 202(1):346–366.

[Ryckelynck, 2009] Ryckelynck, D. (2009). Hyper-reduction of mechanical models involving internal variables. *International Journal for Numerical Methods in Engineering*, 77(1):75–89.

[Ryckelynck et al., 2011] Ryckelynck, D., Benziane, D. M., Cartel, S., and Besson, J. (2011). A robust adaptive model reduction method for damage simulations. *Computational Materials Science*, 50(5):1597–1605.

[Ryckelynck et al., 2006a] Ryckelynck, D., Chinesta, F., Cueto, E., and Ammar, A. (2006a). On the a priori model reduction: Overview and recent developments. *Archives of Computational Methods in Engineering*, 13(1):91–128.

[Ryckelynck et al., 2006b] Ryckelynck, D., Chinesta, F., Cueto, E., and Ammar, A. (2006b). On the a priori model reduction: overview and recent developments. *Archives of Computational methods in Engineering*, 13(1):91–128.

[Ryckelynck et al., 2012] Ryckelynck, D., Vincent, F., and Cantournet, S. (2012). Multidimensional a priori hyper-reduction of mechanical models involving internal variables. *Computer Methods in Applied Mechanics and Engineering*, 225:28–43.

[Saavedra Redlich, 2012] Saavedra Redlich, K. (2012). *Stratégie multiéchelle pour l'analyse du couplage flambage-délaminage de composites stratifiés*. PhD thesis, Cachan, Ecole normale supérieure.

[Schmit and Glauser, 2004] Schmit, R. and Glauser, M. (2004). Improvements in low dimensional tools for flow-structure interaction problems: using global POD. In *42nd AIAA Aerospace Sciences Meeting and Exhibit*, page 889.

[Signorini et al., 2017] Signorini, M., Zlotnik, S., and Diez, P. (2017). Proper generalized decomposition solution of the parameterized helmholtz problem: application to inverse geophysical problems. *International Journal for Numerical Methods in Engineering*, 109(8):1085–1102.

[Simo and Hughes, 2006] Simo, J. C. and Hughes, T. J. (2006). *Computational inelasticity*, volume 7. Springer Science & Business Media.

[Sirovich, 1987] Sirovich, L. (1987). Turbulence and the dynamics of coherent structures. I. Coherent structures. *Quarterly of Applied Mathematics*, 45(3):561–571.

[Smith et al., 2013] Smith, I. M., Griffiths, D. V., and Margetts, L. (2013). *Programming the finite element method*. John Wiley & Sons.

[Souid et al., 2009] Souid, A., Delaplace, A., Ragueneau, F., and Desmorat, R. (2009). Pseudodynamic testing and nonlinear substructuring of damaging structures under earthquake loading. *Engineering Structures*, 31(5):1102–1110.

[Toselli and Widlund, 2006] Toselli, A. and Widlund, O. (2006). *Domain decomposition methods-algorithms and theory*, volume 34. Springer Science & Business Media.

[Van der Vorst, 2003] Van der Vorst, H. A. (2003). *Iterative Krylov methods for large linear systems*. Number 13. Cambridge University Press.

[Vassaux et al., 2015] Vassaux, M., Richard, B., Ragueneau, F., and Millard, A. (2015). Regularised crack behaviour effects on continuum modelling of quasi-brittle materials under cyclic loading. *Engineering Fracture Mechanics*, 149:18–36.

[Veroy and Patera, 2005] Veroy, K. and Patera, A. (2005). Certified real-time solution of the parametrized steady incompressible Navier–Stokes equations: rigorous reduced-basis a posteriori error bounds. *International Journal for Numerical Methods in Fluids*, 47(8-9):773–788.

[Veroy et al., 2003a] Veroy, K., Prud'Homme, C., and Patera, A. T. (2003a). Reduced-basis approximation of the viscous burgers equation: rigorous a posteriori error bounds. *Comptes Rendus Mathématique*, 337(9):619–624.

[Veroy et al., 2003b] Veroy, K., Prud'Homme, C., Rovas, D., and Patera, A. (2003b). A posteriori error bounds for reduced-basis approximation of parametrized noncoercive and nonlinear elliptic partial differential equations. In *16th AIAA Computational Fluid Dynamics Conference*, page 3847.

[Veys, 2014] Veys, S. (2014). *Un framework de calcul pour la méthode des bases réduites: applications à des problèmes non-linéaires multi-physiques*. PhD thesis, Université Joseph Fourier (Grenoble I).

[Vitse, 2016] Vitse, M. (2016). *Model-order reduction for the parametric analysis of damage in reinforced concrete structures*. PhD thesis, Université Paris-Saclay.

[Vitse et al., 2019] Vitse, M., Néron, D., and Boucard, P.-A. (2019). Dealing with a nonlinear material behavior and its variability through PGD models: Application to reinforced concrete structures. *Finite Elements in Analysis and Design*, 153:22–37.

[Wu et al., 2003] Wu, G., Liang, Y., Lin, W., Lee, H., and Lim, S. (2003). A note on equivalence of proper orthogonal decomposition methods. *Journal of Sound and Vibration*.

[Yamamoto and Baker, 2011] Yamamoto, Y. and Baker, J. W. (2011). Stochastic model for earthquake ground motion using wavelet packets. *Bulletin of the Seismological Society of America*, 103(6).

[Zentner et al., 2013] Zentner, I., D'Amico, L., and Cacciola, P. (2013). Simulation of non stationary ground motion compatible with NGA-spectra. In George Deodatis, B. R. E. and Frangopol, D. M., editors, *Safety, Reliability, Risk and Life-Cycle Performance of Structures and Infrastructures*, pages 2349–2355.

[Zerva, 1988] Zerva, A. (1988). Seismic source mechanisms and ground motion models. *Probabilistic Engineering Mechanics*, 3(2):64–74.

[Zienkiewicz et al., 2000] Zienkiewicz, O. C., Taylor, R. L., Taylor, R. L., and Taylor, R. L. (2000). *The finite element method: solid mechanics*, volume 2. Butterworth-heinemann.

[Zlotnik et al., 2015] Zlotnik, S., Díez, P., Modesto, D., and Huerta, A. (2015). Proper generalized decomposition of a geometrically parametrized heat problem with geophysical applications. *International Journal for Numerical Methods in Engineering*, 103(10):737–758.