Modélisation Multi-échelle et Analyse d'Assemblages Macro-moléculaires Ambigus, avec Applications au Complexe du Pore Nucléaire

Tom Dreyfus

UNIVERSITE DE NICE-SOPHIA ANTIPOLIS

ECOLE DOCTORALE STIC

SCIENCES ET TECHNOLOGIES DE L'INFORMATION ET DE LA COMMUNICATION

THESE

pour obtenir le titre de

Docteur en Sciences

de l'Université de Nice-Sophia Antipolis

Mention Informatique

présentée et soutenue par

Tom DREYFUS

Modélisation Multi-échelle et Analyse d'Assemblages Macro-moléculaires Ambigus, avec Applications au Complexe du Pore Nucléaire

Multi-scale Modeling and Analysis of Ambiguous Macro-molecular Assemblies, with Applications to the Nuclear Pore Complex

Thèse dirigée par Frédéric CAZALS

soutenue le 20 décembre 2011

Jury:

M. Joachim Giesen	Professeur, Université de Jena	Rapporteur
M. Patrick Schultz	Directeur de Recherche, IGBMC	Rapporteur
M. Gilles Bernot	Professeur, UNSA	Examinateur
M. Jean-Daniel Boissonnat	Directeur de Recherche, INRIA	Examinateur
M. Alain Denise	Professeur, UPS	Examinateur
M. Felix Rey	Directeur de Recherche, Institut Pasteur et Académie des Sciences	Examinateur
M. Frédéric Cazals	Directeur de Recherche, INRIA	Directeur

Contents

La critique est nécessaire mais l'invention est vitale car en toute invention il y a une critique de la convention.

Gustave Parking

Quoi qu'il arrive, une découverte, une idée lancée, n'appartient plus à son auteur. Galilée s'est récusé, mais la terre n'a pas pour autant cessé de tourner.

Emmanuel Boundzéki Dongala
Un Fusil dans la main, un poème dans la poche

Remerciements

Il y a énormément de monde que je souhaite remercié, aussi bien professionnellement que personnellement, aussi bien mes collègues de travail que mes amis et ma famille. Tous ont participé d'une manière ou d'une autre à l'accomplissement de ma thèse, et à la personne que je suis aujourd'hui.

Mon directeur de thèse, Frédéric, par sa ténacité et sa constante motivation, m'a appris au cours de ces années à transformer la simple intuition en raisonnement détaillé. *Précision, Concision, Exactitude et Originalité*: les mots qu'il a inscrits sur mon tableau il y a presque 4 ans sont maintenant inscrits en moi, même si je suis loin de les respecter tous à la lettre ! Merci.

Mon amour, Andreea, rencontrée au début de ma dernière année de thèse, a non seulement su combler un vide en moi, mais m'a aussi beaucoup soutenu et supporté dans cette dernière année pleine de stress. Ses encouragements m'ont aidé dans autant de directions que j'ai de passions, même lorsque je ne les connaissais pas ! Merci.

Mon meilleur ami et beau frère, Guillaume, m'a apporté autant de choses, de par sa passion, ses connaissances, son ouverture d'esprit et son humour. Alors que je cherchais encore ma voie, il m'a aidé à comprendre ce que je voulais, sans jamais oublier qui j'étais. Il m'a beaucoup aidé à voir le monde tel que je le vois aujourd'hui. Merci.

A tous les autres que je n'ai pas cités, collègues, amis ou famille, et qui j'espère se reconnaîtront:

MERCI

List of Acronyms

NPC:	Nuclear Pore Complex
cryo-EM:	cryo-electron-microscopy
TAP:	Tandem Affinity Purification
TOM:	Toleranced Model
CW VD:	Compoundly Weighted Voronoi Diagram
MCIS:	Maximal Common Induced Sub-graph
MCES:	Maximal Common Edge Sub-graph

List of Notations

List of Figures

List of Tables

Chapter 1

Introduction (English Version)

1.1 Reconstructing Large Systems by Data Integration

1.1.1 Macromolecular Machines and Biological Functions

Biology rests on macro-molecular complexes, so that understanding biological phenomena from the structural standpoint at the atomic level requires describing such complexes. In its most general form, this task remains an open challenge, and numerous sub-questions are faced. The first one is concerned with the stability of complexes, as the life-span of biological complexes varies a lot, from transcient (few micro-seconds, like complexes involved in oxydo-reduction) to obligate (permanent, like some multi-subunit enzymes). The second one is concerned with the specificity of interactions, as the number of partners of a molecule can vary dramatically.

In investigating these questions, a key difficulty is the size of the systems, since the largest assemblies, such as viral capsides or the Nuclear Pore Complex (NPC), may involve hundreds of polypeptidic chains. Size also poses problems in terms of plasticity, as the composition of big assemblies may vary over time — a particular assembly may contain different proteins at different moments of the cell cycle. Another key difficulty is flexibility, as the conformations of the molecules may change upon formation of the complex or assembly, or may also change while the complex is operating—we shall discuss this later in the case of the NPC.

These difficulties motivate research activities which are found at the cross-roads of biophysics, structural biology, and computer science. To improve our understanding of the aforementioned biological functions, one would ideally like to build and animate atomic models of these molecular machines. Experiments are naturally key to these modeling activities, as they provide data which can be used to derive models, and to challenge them. But while atomic models of small complexes can be obtained from X ray crystallography and / or Nuclear Magnetic Resonance, the reconstruction of large assemblies such as molecular motors (cell locomotion), branched actin filaments (muscle contraction), chaperonin cavities (protein folding) or nuclear pore complexes (nucleo-cytoplasmic regulation) is more challenging.

1.1.2 Reconstructing Macromolecular Machines by Data Integration

The modeling challenges arising to reconstruct large assemblies are different from those faced for binary docking, and also from those encountered for intermediate size complexes. For binary complexes, key challenges are currently faced to dock flexible molecules [LW10] and to design discriminatory scoring functions [FO10], as evidenced by the community-wide experiment CAPRI (Critical Assessment of PRotein Interaction), whose focus is the blind prediction of complexes which have been resolved by crystallography. Complexes of intermediate size, on the other hand, are also often amenable to a processing mixing cryo electron microscopy (cryoEM) image analysis and classical docking [LTSW09].
For large assemblies, however, these approaches do not restrain the space of solutions enough. More complex strategies must be resorted to, and one particular strategy of interest is reconstruction by data integration [AFK$^+$08]. In a nutshell, this strategy is reminiscent from NMR and consists of mixing experimental data from a variety of sources, so as to find out the model(s) best complying with the data. This paradigm actually requires three ingredients:

- A geometric model of the system studied. For coarse-grain models, a collection of balls representing protein domains is typically used.
- Various experimental data, shedding complementary light on the system. These data are turned into so-called *restraints*, which when added up define a scoring function measuring the coherence between the model and the data.
- An optimization strategy aiming at finding the most significant local minima of the scoring function.

These ingredients can then be used in a process which iteratively mixes computation and data generation. This process has been used for the reconstruction of plausible models of the NPC [ADV$^+$07a], and is presented on Figure 1.1. In the sequel, we make the three ingredients just presented more explicit.

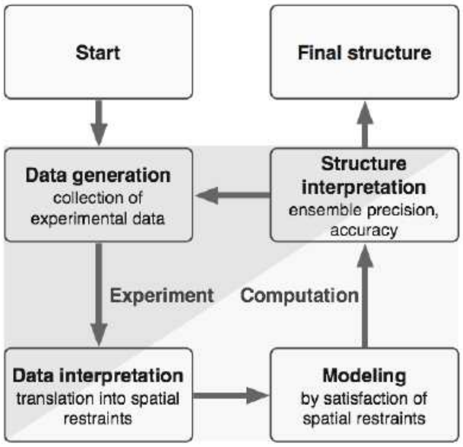

Start
Final structure
Data generation
collection of experimental data
Structure interpretation
ensemble precision, accuracy
Experiment
Computation
Data interpretation
translation into spatial restraints
Modeling
by satisfaction of spatial restraints

Figure 1.1: The four steps of the reconstruction by data integration, applied to the determination of the global structure of the Nuclear Pore Complex. From the Supplementary Information of [ADV$^+$07a].

1.1.3 Relevant Biochemical and Biophysical Data, and Their inherent Difficulties

Turning experimental data into restraints, used to constrain the model being reconstructed, in a non trivial task. Before presenting the restraints used for the NPC, we discuss the relevant experimental data.

Tandem Affinity Purification (TAP) TAP experiments give access to all the protein types found in all complexes containing a prescribed protein type [PCR$^+$01], say P, and can thus be used to constrain the spatial proximity within a model.
More precisely, the method consists of the following steps. First, a fusion protein is created by modifying the gene for P: coding sequences for two *affinity* tags are added, separated by a sequence coding for a protease cleavage site. Upon introducing this engineered gene into a host cell, the modified protein (called PrA-tagged protein) gets expressed and takes its place in its usual complexes—assuming that there is no hindrance induced by the tags themselves. On lysing the cell, the protein complexes containing protein P are retrieved thanks to two affinity purification steps. Each purification step consists of capturing the complexes on an affinity purification column thanks to one of the affinity tags. Between the first and the second purification steps, the complexes hooked on the first column are released by a protease which cuts the linker containing the first affinity tag at the level of the cleavage site. This reveals the remaining affinity tag for the second purification step. Upon completing these purification steps and dismantling the complexes during electrophoresis, one gets a gel with one band per protein type. Mass spectrometry is then used to identify the protein types present.

This list of protein types obtained, also called a pulldown or pulldown, calls for two comments. First, one does not know whether the list of interacting types corresponds a single complex or to several complexes. For example, a list (P, P', P'') obtained by tagging P may correspond to a single complex containing the three species, or to two complexes respectively involving (P, P') and (P, P''). Second, no information on the stoichiometry of protein instances within a complex is available. Despite these inherent combinatorial ambiguities, TAP data are of prime interest for the reconstruction of large assemblies: knowing that protein instances participate in a complex imposes distance restraints between them.

Overlay assays. In contrast with TAP data, overlay assays aim at detecting pairwise protein-protein contacts, allowing to directly constrain protein contacts in the model. A protein P_b called *bait* is first purified and immobilized on nitrocellulose. Then, a fusion protein P_p called *probe* is also purified and overlaid with P_b. After a period of incubation and a washing step for eliminating unbound probes, the detection of overlaid proteins is carried out [Hal04], yielding a signal $S_{p,b}$, which is specific from the protein complex probe-bait.

However, note that the observed signal contains background noise due to the two following reasons: first, during the purification step, contaminant particles may be not eliminated; second, during the incubation step, the number of non specific interactions in an assay increases linearly in time. Extracting the relevant signal is thus a challenging task.

Ultracentrifugation. Ultracentrifugation allows to determine the shape of globular (domains of) proteins, which is useful to assign rough shapes to proteins.
A centrifuge is a refrigerated, evacuated chamber containing a rotor which is driven by an electrical motor capable of high speed rotation. The experiment consists in rotating a protein sample in the centrifuge. Two opposite main forces act on the sample: the centrifuge force and the forces of friction with the solvent. Thus, the sample migrates until it reaches the bottom or the top of its container, or until equilibrium between forces is reached. It is possible to use a solvent with a density gradient (such as sucrose): in this case, the protein sample will migrate to the zone of the solvent sharing the same density, if any. In practice, the density is determined by comparing the sample to a set of marker proteins that migrates to the same zone of the solvent. By measuring the sedimentation velocity of the protein sample, one can determine an abstract value called the sedimentation coefficient S, which is constant among the marker proteins having the same density. S is directly related to the molecular mass, the volume and the shape of the involved proteins [Eri09]. Intuitively speaking, small values of S correspond to elongated proteins having a large surface subjected to forces of friction, while large values of S to globular proteins.

Cryo-electron microscopy (cryoEM). A further approach under active development is cryo-electron microscopy (CryoEM) [Fra06]. Structures as large as whole cells and as small as individual proteins can be imaged with electrons, and with cryo techniques final resolutions on the order of 0.3 nanometers have been attained. In single particle analysis, bombarding isolated samples with electrons yields images corresponding to different viewpoints, and these can be combined into a 3D model of the particle. In cryoEM tomography, a given sample is instead bombarded at incremental degrees of rotation, from which a 3D model can also be reconstructed. In both cases, the result is a 3D density map, where each voxel encodes the density of matter. This density is in general very noisy due to the low electron doses used to avoid damaging biological specimens. Choosing a density level for contouring a surface (called the envelope) enclosing the model is non-trivial, as the intensity is generally high for globular domains of the proteins, but low for unstructured regions such as linkers connecting these domains. Typically, medium (around 5Å, secondary structure elements visible) to low (less than 10 Å, domains visible) resolutions are achieved in cryoEM. In favorable cases, fitting existing and/or modelled structural elements into such maps yields atomic resolution models.

Immuno-electron microscopy. In immuno-EM, one wishes to locate specific proteins within an assembly [SH01], this positional information being used to favor the location of proteins in the model.
To this end, the protein of interest is attached to a specific antibody or a big tag. The detection of the shape of the antibody or the additional mass of the tag allows to locate the protein in the cell. In the NPC case, antibodies against the required antigen are labeled with gold particles and are then examined under the electron microscope. Then regions of images containing assemblies with gold-labeled particles are selected with circles: the center of each circle is manually aligned with the assembly. After a quality selection step, batches of assembly images are manually aligned to generate montages; the position of every gold particle in each montage is measured.

A major issue from immuno-localization comes from the impossibility to establish precisely the coordinates of the gold particles. Moreover, for several reasons (rotation of the antibodies around the tags, distortion or damage of the samples during preparation), each montage demonstrates a high degree of gold particle scattering.

Except for the overlay assays, biochemical and biophysical data cannot be directly interpreted as simple geometric restraints. For example, pulldowns obtained with the TAP do not reveal pairwise interactions between proteins, inducing a combinatorial aspect to the geometrical restraint. The variety of geometrical restraints included in the scoring function is also important: there may be a bias in the final model if geometrical restraints are not independent (e.g. geometrical restraints on the locations and distances between several proteins), or if a particular feature (e.g. locations of proteins) is represented by several geometrical restraints. All these issues imply that the models computed inherently encode uncertainties that are very hard to interpret. Even if a high quality model is produced from this methodology, it is impossible to make precise quantitative statements about the shape of proteins, or about their relative positions and contacts.

1.2 The Nuclear Pore Complex: a Concise Description

Biological features. The Nuclear Pore Complex (NPC for short) is the largest protein assembly known to date in eukaryotic cells. It is involved in the transit of molecules across the nuclear envelope, see Figure 1.2, with in particular the import of proteins or the export of RNA [WR10].

The NPC is formed by a channel of circa 100 nm of diameter, filaments containing docking sites for molecules crossing over the channel, and a basket on the nuclear side. Small particles ($<$30kDa) are able to pass through the NPC by passive diffusion, but larger particles may be recognized by the filaments containing specific sequences called the FG-repeat sequences [DPU$^+$03], which help in their active transport from one side to the other. The proteins involved in this process are known as karyopherins.

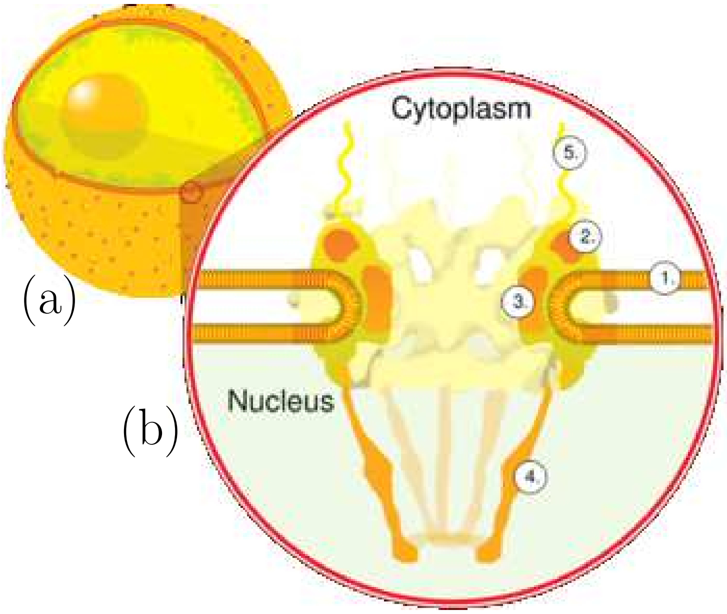


Figure 1.2: Sketchy structure of the Nuclear Pore Complex (a) The NPC are located on the nuclear envelope. (b) Zooming on the nuclear envelope: (1)Nuclear envelope (2) Outer ring (3) Spokes (4) Basket (5) Filaments. Picture from Wikipedia at `http://en.wikipedia.org/wiki/Nuclear_pore`.

Structural features. Experiments have shown that the NPC is ring-shaped with a 8-fold rotational symmetry axis perpendicular to the nuclear envelope plane [ADV$^+$07b]. It is thus made of 8 identical blocks termed spokes, see Figure 1.3. Each spoke has a nuclear side and a cytoplasmic side, each of them termed half-spoke.

To describe models of the NPC, it is convenient to talk about protein *types* and protein *instances* i.e. copies of a given type. In a recent work [ADV$^+$07b], the NPC has been modeled using 30 different protein *types*, with 29 protein instances of 27 different protein types for a cytoplasmic side half-spoke, and 28 protein instances of 25 different types of protein for a nuclear half-spoke. Therefore, these models involve a total of $8 \times (29 + 28) = 456$ proteins instances.

From a global standpoint, the NPC may be segmented in four functional concentric cylinders [HSBH07], namely (i) the channel cylinder, containing protein types having unstructured regions i.e. filaments regulating the active transport; (ii) the adapter cylinder, involving intermediate protein types between channel protein types and scaffold protein types; (iii) the coat cylinder, which defines the scaffold of the NPC; (iv) the pore membrane cylinder, anchoring the NPC into the nuclear membrane.

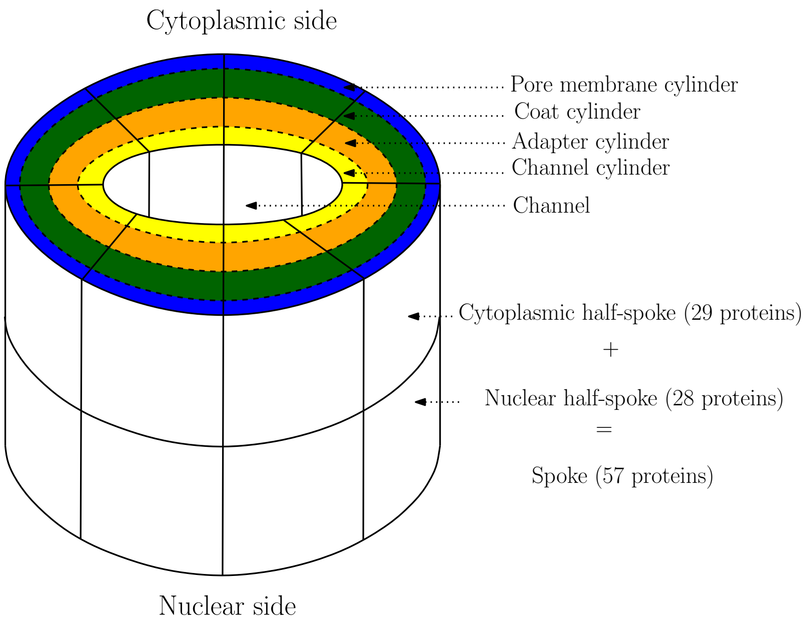


Figure 1.3: Schematic representation of the structure of the Nuclear Pore Complex. It is composed of 8 symmetrical spokes, each of them divided into two symmetrical half-spokes. Each spoke contains 57 proteins of 30 different types located in the Channel (yellow), Adapter (orange), Coat (dark green) or Pore membrane (blue) cylinder. Adapted from [HSBH07].

1.3 Modeling the Nuclear Pore Complex

In [ADV$^+$07a, ADV$^+$07b], Alber et al proposed the first coarse grain model of the whole yeast NPC based on data integration. The reconstruction procedure was used to select $N = 1000$ plausible models. We now briefly review the three steps of the reconstruction procedure mentioned in Section 1.1.2, and also briefly discuss the exploitation of the N models.

1.3.1 A Hierarchy of Coarse-grain Models

A hierarchy of models. In the model of [ADV$^+$07a], the NPC is represented with a 4-level hierarchy, see Figure 1.4:

- (i) the assembly level, a given model being denoted $A_i, i = \{1,\ldots,N\}$, where N is the total number of models;
- (ii) the half-spoke unit U_s^θ, where $s = 1,\ldots,8$ in the spoke index, and the index $\theta \in \{1,2\}$ refers to the cytoplasmic and nuclear sides, respectively.
- (iii) the protein instance level, a given instance being denoted $p_j, j = \{1,\ldots,456\}$.
- (iv) the bead / particle level, a particular bead B_k being parametrized by its center and radius.

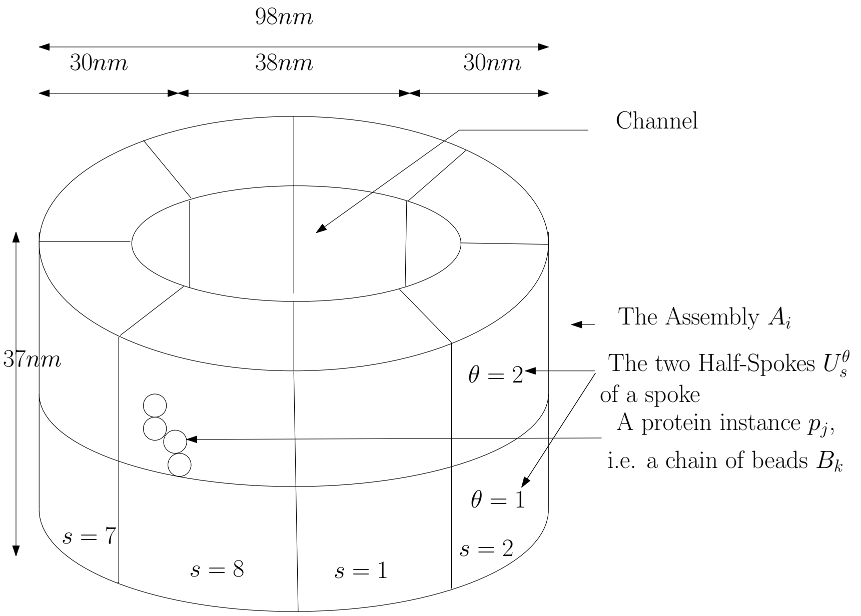


Figure 1.4: The hierarchical representation of the NPC: (i) the overall assembly A_i (ii) the half-spoke unit U_s^θ (iii) the protein instance p_j (iv) the bead B_k.

Before running the optimization process, all the instances of a protein type have the same geometry, which consists of several beads of a fixed size. (Note that this size depends on the protein type.) The goal of the optimization is to reshape these initial geometries, to comply with the experimental data.

To account for all possible inter-spoke interactions between protein instances without FG-repeat sequence, the authors consider that each half-spoke has four neighboring half-spokes that are obtained by rotating of $2\pi/8$ all cytoplasmic half-spokes. (For proteins with FG-repeat sequences, the authors supposed that filaments could interact with protein instances that are not in the neighboring half-spokes.)

Finally, the NPC is anchored on the Nuclear Envelope, which is a mould for the assembly. The Nuclear Envelope is represented by spheres with a fixed diameter corresponding to the average thickness of the NE (4.5nm).

Representation of the protein instances. In order to consider interactions and locations of the different regions of a protein, the authors introduced nine levels of representation numbered from $k = 1$ to 9 – see Figure 1.5. At the level $k = 1$, one finds the root representation, corresponding to the finest representation of the protein. All other representations are derived from the first level and allow to consider independently the following features of a protein. The classification goes as follows:

- representation level $k = 2$ and $k = 3$: the globular domains,
- representation level $k = 4$: the unstructured regions,
- representation level $k = 5$: the non membrane-spanning regions,
- representation level $k = 6$: the membrane spanning regions of pore membrane proteins,
- representation level $k = 7$: the perinuclear non membrane spanning regions of pore membrane proteins,
- representation level $k = 8$: the pore side non membrane spanning regions of pore membrane proteins,
- representation level $k = 9$: the C-terminal region of Pom152, a pore membrane having homotypic interactions at the medial plane of the NPC through its C-terminal region.

The root representation consists of a linear flexible oriented chain of identical beads that fits within an ellipsoid computed from the observed sedimentation coefficient of the protein: the first bead represents the C-terminal region of the protein and the last one represents the N-terminal region of the protein.

The Cartesian coordinates of a bead at representation levels $k = 2$ to $k = 9$ has the same coordinates as a bead in representation $k = 1$, with two exceptions. For $k = 3$ and $k = 6$, a single bead is used, its coordinates being computed from a weighted average of the coordinates of the beads at the representation level $k = 2$ and $k = 1$, respectively.

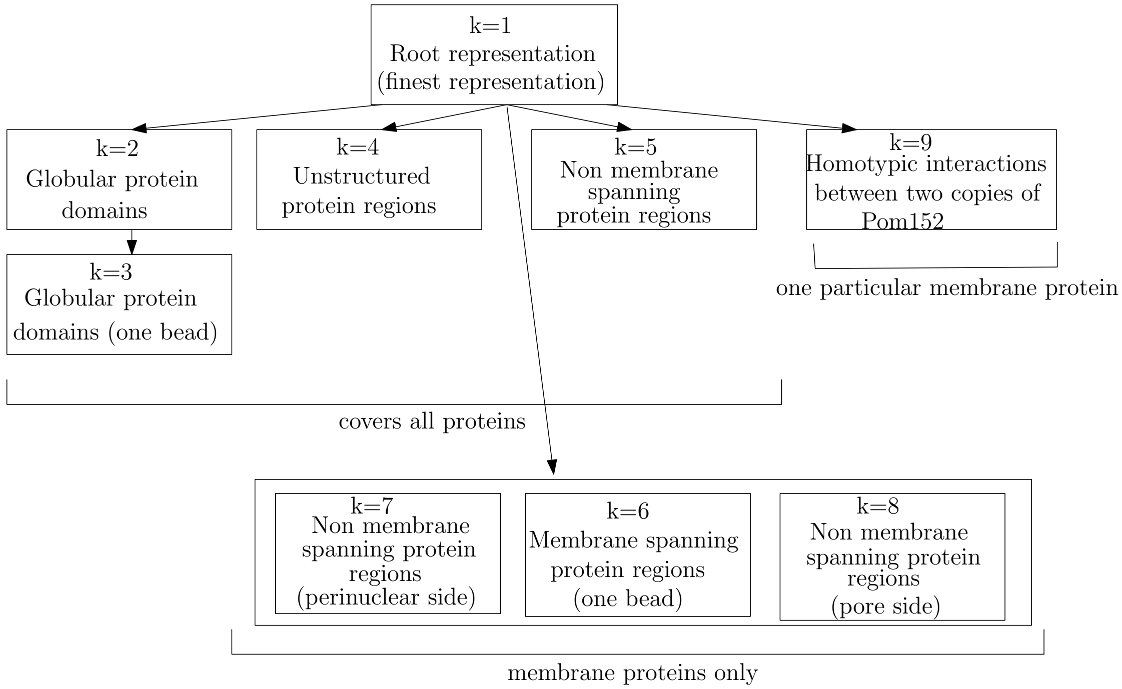


Figure 1.5: The 9 representation levels of proteins in the model of Alber et al. The arrow from $k = 2$ to $k = 3$ corresponds to coarsening the protein representation. All the remaining arrows can be seen as selectors, since they discriminate a particular region / feature of the protein. It should also be noticed that there are two cases where a single bead is enforced, namely for $k = 3$ and $k = 6$.

1.3.2 Input Data and the Associated Restraints

The scoring function is interpreted as a sum of spatial restraints, a restraint being a real-valued function defined at some representation level k, encoding the coherence of the model with respect to some experimental data. The function returns 0 if the model is coherent with the data, and some positive value otherwise. There are three types of spatial restraints :

- the localization restraints, which constrain the position of some beads;
- the distance based restraints, which constrain the distances between two beads;
- the symmetry restraints, which constrain two sets of beads to have the same property;

We now discuss these restraints as a function of the available data.

Tandem Affinity Purification (TAP)

Practical matters. For a tagged protein type, the authors obtained several pulldowns with a quality factor ranging from 1 (top quality) to 3 (low quality). The authors reported 75 pulldowns containing from two to 20 protein types, classified as follows: top quality: 34; intermediate quality: 21; low quality: 20.

Restraints. Restraints related to a pulldown are used at the representation levels $2, 4$ and 9. Recall that a pulldown cannot be directly interpreted as a spatial restraint due to its combinatorial nature. For each half-spoke U_s^{θ}, a two step strategy is applied for selecting the pairs of protein instances with types in the pulldown for which a contact is constrained. First, a complete weighted graph is defined over the node set corresponding to the protein types. The weight of an edge corresponding to the types P and P' is set as follows: a score is computed for each pair of protein instances p_i and p'_j of the types P and P', such that at least one of the two protein instances is in U_s^θ and the other one is in the same half spoke or one of the four neighbor half-spokes. (As already mentioned, the selection of protein instances with a protein with a FG-repeat sequence is not restricted to neighbor half-spokes.) This score is the minimal distance between centers of beads of p_i and p'_j. Then the pair of protein instances minimizing this score is retained and the weight of the edge (P,P') is set to this score. Second, a minimal spanning tree is computed on the complete graph. The selected edges define a minimal connectivity amongst the protein types. The sum of the weights of selected edges is the score of the spatial restraint of the pulldown.

Overlay assays

Practical matters. The authors generated 30 overlay assays corresponding to the 30 probe protein types involved in the NPC model: they obtain a matrix of signals $S_{p,b}$ for each possible pairs of probe / baits. For reducing the noise-to-signal ratio, the authors first normalized the signal $S_{p,b}$ from each bait (b) with a given probe (p) to the general background signal across all n baits for that probe (on one overlay blot). A similar correction was applied for the general background signal for each bait across all k probes (on the different overlays) to generate a normalized signal [1] for each bait with each probe $Y_{i,j}$:

$$Y_{i,j} = \frac{S_{i,j}}{\left(\sum_{i'=1}^{i'=n} S_{i',j} + \sum_{j'=1}^{j'=k} S_{i,j'} - S_{i,j}\right)} \tag{1.1}$$

Then, only probe-bait interaction giving a signal more than 10 times above its average was considered significant. Following this protocol, only 7 pairwise interactions (involving 5 protein types) are qualified as significant.

Restraints. Restraints related to overlay assays are used for representation levels $2,4$ and 9. The pairwise interactions observed with the overlay assays are constrained using the same strategy as the one used for pulldowns. Note, though, that the minimal spanning tree of a connected graph with two nodes is the graph itself.

Ultracentrifugation

Practical matters. The authors use ultracentrifugation to determine the finest representation of a protein in terms of beads. Each protein instance is represented by a prolate ellipsoid (i.e an ellipsoid rotated about its major axis). The number of beads per protein is set to the nearest integer value of the axial ratio (a/b) of the prolate ellipsoid and the radii of the beads are scaled to reproduce the volume of the protein, estimated from its sequence. To compute the axial ratio, the authors derive from the sedimentation coefficient S of a protein type the Perrin function P, which is a molecular shape function. Using the inversion formula of a prolate ellipsoid with a development of order n (n is generally set to 6) [HC95], one can obtain the following formula:

$$(a/b) = a_0 + \sum_{i=1}^{i=n} a_i P^i \tag{1.2}$$

where a_0 and $(a_i)_{i \le n}$ are given parameters [HC95, Table 3 and 4].

The authors also use ultracentrifugation for estimating the length D of the largest axis of the ellipsoid fitting the Y-complex, a well known protein complex composed of 7 protein types—see details in Section 4.2.1.

Restraints. Three spatial restraints depend only on ultracentrifugation data: the first one constrains any pair of beads to have an empty intersection; the second one constrains two consecutive beads of a protein instance to be tangent; the last one constrains beads of protein instances of the Y-complex in a half-spoke to be contained in a sphere of diameter D.

[1] In fact, the formula provided in [ADV$^+$07a, Supplemental, page 11] is as follows:

$$Y_{i,j} = \frac{S_{i,j}}{\sum_{i'=1}^{i'=n} \sum_{j'=1}^{j'=k} S_{i',j'}}.$$

That is, the signal is divided by the sum of all signals, that is constant among all signals. We believe that this do not normalize the signals

Cryo-electron microscopy

Practical matters. The map obtained from single particle cryoEM experiments were used to derive several important global features: the eight-fold (eight spokes) and two-fold (two half-spokes per spoke) symmetries, the anchoring of the NPC into the nuclear envelope, the diameter of the channel. Also, the density map is used as reference for maps generated by immuno-electron microscopy.

Restraints. Spatial restraints derived from cryoEM are used in two ways. First, a spatial distance restraint for the representation level 5 constrains the beads of protein instances and the beads of the nuclear envelope to have an empty intersection. Second, due to the eight-fold and two-fold symmetries, two spatial symmetry restraints for the representation level 2 constrain two protein instances (p_i, p_j) of type P present in two different half-spokes to share the same configuration of beads: (i) by constraining two pairs of beads of p_i and p_j to have an identical distance, and (ii) by constraining two sets of four beads of different protein instances in two different half-spokes to share the same dihedral angle.

Immuno-electron microscopy

Practical matters. Except for two nucleoporins (Nup1, and Nup60), the authors collected 300 particle positions for each protein type, by aligning the selected circles drawn on electron microscopy images, with the intersection point of the central Z-axis and equatorial plane of the NPC. To account for the inherent scattering, the authors created a scenario where the position of proteins was known –the model distribution. More precisely, a plastic membrane is coated with tagged proteins, is observed, and a Gaussian distribution is fitted so as to recover the model distribution. Getting back to the NPC experiments, this Gaussian distribution is slided along the Z and R axis so as to maximize the overlap with the experimental distribution of the observed gold particles. The overlap is defined as:

$$Overlap = 1 - \frac{1}{2}\sum_{i=1}^{N_b} |E_i - C_i| \tag{1.3}$$

where E and C are respectively the experimental and calculated distributions normalized to add up to 1, and N_b is the number of the 2.5nm bins spanning the distributions from 0 to a large positive value.
Finally, the authors correct for localization errors presumably due to the steric hindrance of the nuclear envelope and of the NPC. These structural portions are embedded in the structure of the NPC as visualized in reference cryoEM maps. Hence, the total immuno-electron microscopic map is manually aligned to maximize its overlap with the cryo-electron microscopic map.

Restraints. Restraints related to immuno-electron microscopy are used for the representation level 1. Two spatial localization restraints are defined to favor Z and R coordinates of all beads of all proteins instances (except Nup1 and Nup60) to be in the described range.

1.3.3 Optimization Methods

As mentioned in Section 1.3.1, a protein is represented by a collection of beads. In the sequel, by *configuration* of the NPC, we refer to the position and the size of these beads at the root representation level. If we assume that the radii of all balls can be estimated so as to match the volume of a protein estimated from its sequence, notice that the dimension of the configuration space is exactly three times the number of balls.

Overall strategy. The optimization strategy consists of the two steps illustrated on Figure 1.6. The first one selects 200,000 configurations. The second one starts with the top 20,000 configurations of the first round, and refines them. Finally, the configurations corresponding to the top 1000 scores are retained. The optimization of a configuration A_i of the NPC is done by minimizing a scoring function $f(A_i)$ which is a linear combination of a subset of the spatial restraints. Indeed, the restraints are not considered all at once, but are instead gradually added along the optimization [BG85]. The minimization of $f(A_i)$ is done using rounds of conjugate gradients and molecular dynamics with simulated annealing. For each configuration A_i, ten rounds are realized during the coarse grain step, and at most eight rounds during the fine grain step.

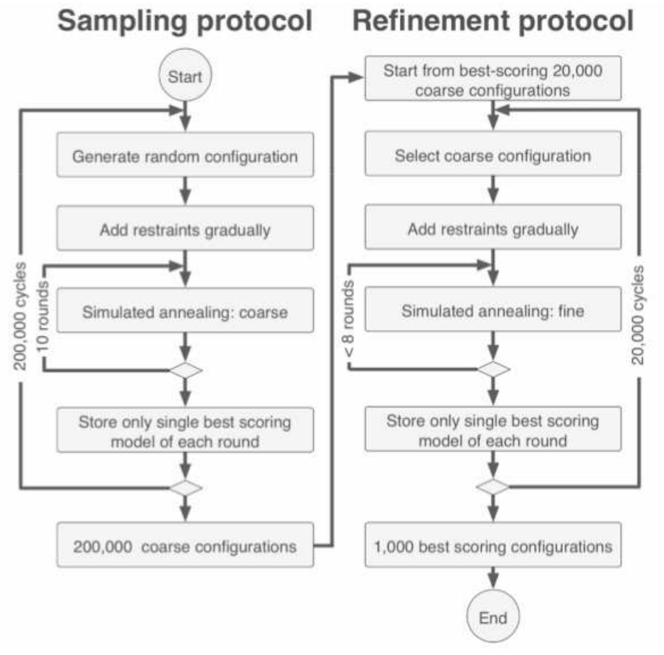

Sampling protocol
Refinement protocol
Start
Generate random configuration
Add restraints gradually
Simulated annealing: coarse
Store only single best scoring model of each round
200,000 coarse configurations
200,000 cycles
10 rounds
Start from best-scoring 20,000 coarse configurations
Select coarse configuration
Add restraints gradually
Simulated annealing: fine
Store only single best scoring model of each round
1,000 best scoring configurations
End
< 8 rounds
20,000 cycles

Figure 1.6: The two main steps of the optimization protocol: coarse optimization (left), and fine optimization on the (right). From the Supplemental Figure 11 of [ADV$^+$07a].

1.3.4 Results and their Assessment

The $N = 1000$ structures were not released, but two derived products were so: contact frequencies, and probability density maps. These are available from `http://salilab.org/npc/`. We now discuss briefly these results.

Contact frequencies significance. The *contact frequency* of a pair of protein types (P, P') is the fraction of the selected models, out of $N = 1000$, which display a contact between at least one instance p of P and one instance p' of P'. That is, a particular model qualifies provided that there is at least one pair of beads, one bead from p and one from p', such that the distance between their centers satisfies a distance constraint. Note that if two particular protein instances of types P and P' are in contact in all models, but all other possible pairs of protein instances of these types are not, the contact frequency is equal to 1. That is, contact frequencies do not account for the stoichiometry of the contacts.

Probability density maps uncertainties. To exploit the $N = 1000$ models selected globally, the author computed one *probability density map* per protein type. More precisely, consider one particular protein type. Having collected all the balls of the protein instances of this type across the N models, the authors blended these balls to produce a probability density map representing the overage position of all these instance.

Contouring this maps yields a representation of the proteins, as illustrated on Figure 1.7. These density maps offer a qualitative result on shapes and locations of proteins in the NPC. However, due to uncertainties on biochemical and biophysical data, and bias introduced by the methodology, it is very hard to make quantitative analysis on these maps. As an extremal case, the ring around the NPC on Figure 1.7 corresponds to the density map of the 16 protein instances of Pom152.
The following quotes witness the qualitative nature of the analysis carried out [ADV$^+$07a, ADV$^+$07b]:
$\ll$ *Our map is sufficient to determine the relative positions of proteins in the NPC; we do not interpret features smaller than this precision.* $\gg$
$\ll$ *Because of the limited precision of the information used here, the localization volume of a protein should not be mistaken for its density map, such as that derived by cryo-EM.* $\gg$
$\ll$ *The localization volumes [...] allow a visual interpretation of the relative proximities of the proteins.* $\gg$

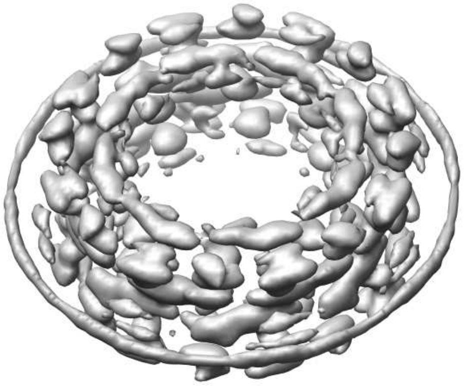


Figure 1.7: A level set surface corresponding to the threshold 0.5 within the 3D probability density map resulting from the NPC reconstruction. Plotted with the CHIMERA software.

1.4 Voronoi Diagrams and α-complexes

1.4.1 Handling Uncertainties with Affine Voronoi Diagrams and Related α-shapes

Voronoi diagrams, spatial partitions, and parameterized models As we have seen while presenting the reconstructions of the NPC, the uncertainties on the input directly translate on the output. In particular, given a probability density map, placing the protein instances is an ill posed problem which, in general, does not have a unique solution. On the other hand, in a number of cases, one can delimit a zone within which interesting phenomena can be confined. This zone can then be used as the support of a parameterized model sweeping the region of interest. Naturally, confining the model in the region may prevent finding features of the model. But in a number of cases, delimiting the region is rather straightforward. The argument which will be refined when dealing with probability density maps will be to use to same uncertainty for the zone (and also for the parameterized model) as that observed on the input data.

While the previous paradigm seems appealing, the key question consists of defining the parameterized shapes. To this end, a design choice consists of considering a parametrization that yields nested shapes. The classical case in computational geometry is that of the weighted α-shape associated with a power diagram, a remarkable construction proposed by E. Edelsbrunner [Ede92]. For the sake of clarity, we present the even simpler case of α-shapes related to the Euclidean Voronoi diagram.

Euclidean Voronoi diagrams of a set of points. Formally, given a set $\mathscr{S}$ of n points in the Euclidean space $\mathbb{E}$, the Euclidean Voronoi diagram of $\mathscr{S}$ is the partition of the space into *Voronoi cells* such that each Voronoi cell contains all points of the space closer to a point $x_i \in \mathscr{S}$ than any other point $x_{j\neq i} \in \mathscr{S}$ w.r.t. the Euclidean distance:

$$Vor(x_i) = \{x \in \mathbb{E} \text{ such that } \mid\mid x - x_i \mid\mid \leq \mid\mid x - x_j \mid\mid, \forall x_j \neq x_i\}. \tag{1.4}$$

In 2D, all points equally distant from two points $x_i \in \mathscr{S}$ and $x_j \in \mathscr{S}$ are located on a straight line called the *bisector*. The intersection of the Voronoi cells of x_i and x_j, yields a connected portion of the bisector called a *Voronoi edge*. The intersection of two Voronoi edges (or generically of three Voronoi cells) yields a point that is called a *Voronoi vertex*. Let v be a Voronoi vertex. Since v is equally distant to the three points $(x_i, x_j, x_k) \in \mathscr{S}$, it is the center of the ball B circumscribing the triangle (x_i, x_j, x_k). Furthermore, since v is closer to x_i, x_j, x_k than any other point in $\mathscr{S}$, the ball B does not contain any other point of $\mathscr{S}$ in its interior. This property is called the *empty ball* property or *conflict-free* property and can be extended to points on a Voronoi edge or a Voronoi cell.

Dual structure of Euclidean Voronoi diagram. The Voronoi diagram has a dual structure called the Delaunay triangulation, which is obtained by collecting vertices, edges and triangles as follows:
– a Voronoi cell associated with a point x_i contributes a Delaunay vertex;

– a Voronoi edge $e_{i,j}$ associated with two points x_i and x_j contributes the Delaunay edge (x_i,x_j), obtained by joining x_i and x_j;
– a Voronoi vertex $v_{i,j,k}$ associated with three points x_i,x_j and x_k contributes the dual triangle (x_i,x_j,x_k), which may be seen as the convex hull of the three points.

Generically, i.e. of no four points are co-circular in 2D, this dual structure is a triangulation. Due to the empty ball property, the circumscribed spheres of all dual triangles have no point of $\mathscr{S}$ in their interior. Similarly, for each point contained in the interior of a Voronoi edge, there exists an empty sphere passing through the vertices of the dual Delaunay edge. Also, note that the circumscribed ball $B_{i,j}$ of a dual edge (x_i,x_j), namely the ball whose diameter is (x_i,x_j), is empty iff the center of $B_{i,j}$ is located on $e_{i,j}$. A dual edge having this property is said *Gabriel*.
The triangulation is actually a so-called simplicial complex, namely a collection of simplices such that any two of them intersect on a common face if they intersect at all. With this terminology, the Delaunay triangulation consists of k-simplices, where k is the dimension of the simplex: a Delaunay vertex is a 0-simplex; a Delaunay edge is a 1-simplex; a Delaunay triangle is a 2-simplex. The dual of a k-simplex is a Voronoi cell ($k = 0$), a Voronoi edge ($k = 1$) or a Voronoi vertex ($k = 2$): we call Voronoi $2-k$-face the dual of a k-simplex.

Space filling diagrams, the α-complex and the α-shape. The Voronoi diagram admits the following intuitive construction: start growing balls centered on the points, at the same speed[2] $\sqrt{\alpha}$, and they will meet on their common bisectors. The domain covered by these growing balls is called a space filling diagram. Let the restriction of a ball be the intersection between the ball and its Voronoi region. The strategy just presented actually consists of tracking intersections between restrictions. Note that this is a more stringent condition that tracking intersection between balls, as illustrated by the growth of three balls forming an obtuse triangle, see Figure 1.8.

Similarly to the construction of the Delaunay triangulation, we can report simplices corresponding to the intersection of restrictions. All such simplices form another simplicial complex called the α-complex, and the domain of these simplices (the union of the geometric domains of the vertices, edges and triangles) defines the so-called α-shape.

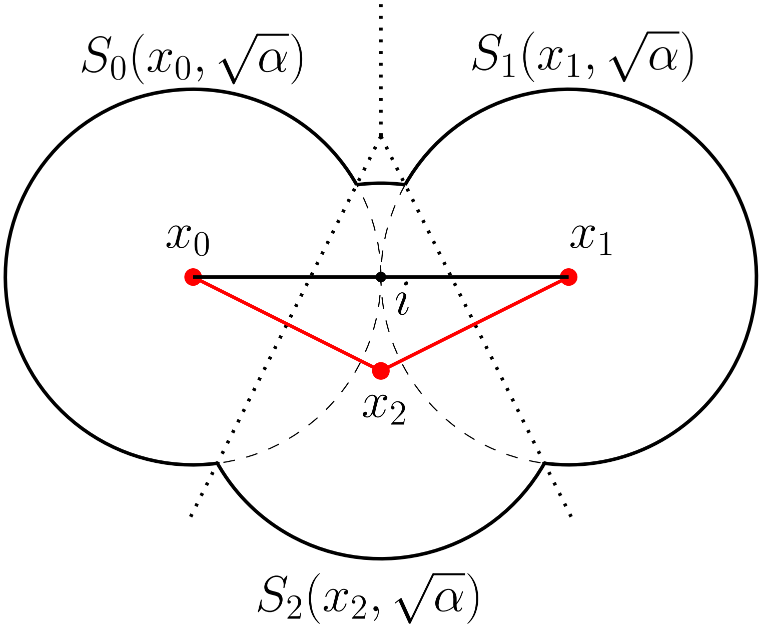


Figure 1.8: A 2D example of the α-complex of three points x_0,x_1 and x_2, for $\alpha > 0$. The Voronoi diagram is drawn in dotted lines, and the Delaunay triangulation in solid lines. The red points and red lines correspond to simplices in the α-complex. The restriction of spheres $S_0(x_0,\sqrt{\alpha}),S_1(x_1,\sqrt{\alpha})$ and $S_2(x_2,\sqrt{\alpha})$ are shown in black solid lines. S_1 and S_2 are drawn in dashed lines. Note that S_0 and S_1 intersect at the point i in the interior of the ball bounding by S_2: the corresponding dual simplex is not Gabriel.

[2] The justification of the speed $\sqrt{\alpha}$ rather than α will be clear when discussing power diagrams.

Note that growing balls is a monotonic process: when the radii are being enlarged, the restrictions are nested, and so are the α-complexes. In fact, the 0-complex matches the collection of points, and for a large enough value, the α-complex matches the whole Delaunay triangulation.

Note also that the monotonic growth process just presented matches the nestedness requirement discussed in Section 1.4.1. In fact, the methodological contribution of this thesis is a generalization of the α-complex. To position this contribution, we now briefly present three classical Voronoi diagrams.

1.4.2 Generalized Voronoi Diagrams

A Voronoi diagram being a partition of the ambient space induced by a (generalized) distance to objects, a variety of such diagrams can be defined. Three of them are particularly interesting, and their specifications are summarized in Table 1.1. In all these cases, the generalized distance has the form

$$a_i \mid\mid x - c_i \mid\mid^k - b_i, \tag{1.5}$$

where the parameters a_i and b_i are respectively called the multiplicative and additive parameters, and $k \in \mathbb{N}^*$ is the power of the distance. The reader is referred to [BWY06] for more on these diagrams.

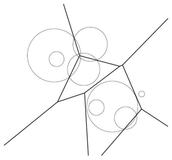
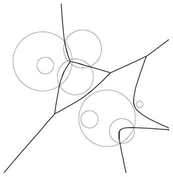
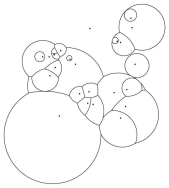
Name	Generalized Distance	Bisector	Diagram in 2D
Power Diagram of spheres: $S_i(c_i, w_i = r_i^2)$	$d(S_i(c_i, w_i), x)$ $=$ $\mid\mid c_i - x \mid\mid^2 - w_i$	Radical hyperplane	
Apollonius Diagram of spheres: $S_i(c_i, r_i)$	$d(S_i(c_i, r_i), x)$ $=$ $\mid\mid c_i - x \mid\mid - r_i$	Hyperboloid or straight lines	
Möbius Diagram of compoundly weighted points: $W_i(c_i, a_i, b_i)$	$d(W_i(c_i, a_i, b_i), x)$ $=$ $a_i \mid\mid c_i - x \mid\mid^2 - b_i$	Hypersphere (possibly degenerated)	

Table 1.1: The classical Voronoi diagrams.

Power diagram. This diagram deals with spheres and the so-called power distance. In the sequel, the dimension of the ambient space is denoted by d, and the objects considered are called spheres. This diagram generalizes the Euclidean case, as all affine Voronoi diagrams are power diagrams. The bissector of two spheres is called a radical flat. Any point located on a $d-k$-dimensional Voronoi face is the center of a sphere which is orthogonal [3]. Two interesting situations not encountered in the Euclidean case may appear: A Voronoi cell of a sphere $S_i \in \mathscr{S}$ (i) may not contain the center of S_i (the dual of the Voronoi cell is not Gabriel) or (ii) may be empty (S_i is *hidden*).

Apollonius diagram. It is the generalization of the Euclidean case to spheres, under an additive distance. (The additive distance refers to the distance of a point to the sphere, rather than the distance to its center. That is, the

[3] Two spheres $S_i(c_i, r_i)$ and $S_j(c_j, r_j)$ are orthogonal provided that $\mid\mid c_i - c_j \mid\mid^2 = r_i^2 + r_j^2$.

radius acts as additive parameter.) Points on the bisectors of k spheres in $\mathscr{S}$ are locii of spheres tangent to the k spheres. For example in 2D, a bisector is either one branch of a hyperbolae, or a straight line. A Voronoi cell of a sphere $S_i \in \mathscr{S}$ (i) always contains the center of S_i (its dual is always Gabriel) and (ii) is empty iff there is another sphere $S_j \in \mathscr{S}$ that contains S_i. Note that two spheres $(S_i, S_j) \in \mathscr{S}$ may have several common Voronoi edges, and that three spheres $(S_i, S_j, S_k) \in \mathscr{S}$ may have at most two common Voronoi vertices.

Möbius diagram. It is the generalization of the power case to compoundly weighted points and compoundly weighted power distance. For two spheres, the bisector is a sphere or a straight line. A Voronoi cell of a compoundly weighted point $W_i \in \mathscr{S}$ (i) has not necessarily a Gabriel dual simplex and (ii) may be empty. Note that one weighted point $(W_i) \in \mathscr{S}$ may have several Voronoi cells, two weighted points $(W_i, W_j) \in \mathscr{S}$ may have several common Voronoi edges and three weighted points $(W_i, W_j, W_k) \in \mathscr{S}$ may have at most two common Voronoi vertices.

1.5 Thesis Overview

From a practical standpoint, this thesis aims at correcting the limitations of the models mentioned in Section 1.3.4. From a methodological standpoint, this goal motivated the development of algorithms and concepts related to curved Voronoi diagrams. We now put these contributions in perspective.

1.5.1 Compoundly Weighted Voronoi Diagram and their λ-Complex

In order to model macro-molecular assemblies with uncertainties, we introduce *toleranced models* to accommodate the positional and conformational uncertainties of protein instances within large assemblies. A toleranced model is a collection of *toleranced balls*, each such ball consisting of two concentric balls called the *inner* and *outer* balls, respectively meant to encode high and low confidence regions. A linear interpolation of the radius of a toleranced ball defines a growth process which is that of the so called *compoundly weighted Voronoi diagram*, or CW for short.

In Chapter 3, we first introduce toleranced models. We then generalize the empty ball property, examine properties of the bisectors, of the Voronoi diagram itself, of its dual structure. We proceed with the generalization of the α-complex, which we call the λ-complex. Finally, we present a naive output sensitive algorithm for computing an abstract representation of the CW VD and its λ-complex.

1.5.2 The Nuclear Pore Complex: Material and Methods

The Nuclear Pore Complex is the largest known macro-molecular assembly in the eukaryotic cell, and understanding its structure and function is a key endeavor in cell and structural biology. In particular, analyzing the location and the partners of each protein instance is required to determine the structure of the assembly. The model of the NPC of Alber et al presented in Section 1.3 is a first step, but the qualitative results makes the interpretation a very hard task.

Chapter 4 provides an introduction to the key features of the NPC. We first introduce three important sub-complexes of the NPC (namely the Y-complex, T-complex and Nup82-complex), which play key roles from the structural and functional standpoints. We then present a brief assessment of the probability density maps of [ADV$^+$07a]. Finally, we present algorithms to construct toleranced model of the NPC based on these probability density maps.

1.5.3 Assessing Pairwise Contacts at the Assembly and Sub-Complex Levels

Chapter 5 presents certain analysis on the NPC, based on the toleranced models presented in the previous chapter, and on the λ-complex of this model. The overall goal is to present a multi-scale analysis of contacts between protein types, as well as the investigation of the protein complexes involving selected protein types.

First, we present the *Hasse diagram* of a toleranced model, that is an interpretation of the λ-complex in terms of protein instances. We use the protein contact history contained in the Hasse diagram to define the contact curve of two protein types, a curve containing stoichiometry dependent information on the contacts between instances of the two types. Finally, focusing on the complexes involving specific types, we present analysis aiming at assessing the symmetry properties of the toleranced model. This particular task is specifically carried out on the three sub-complexes described in Chapter 4.

1.5.4 Assessing Graphical Models of Sub-Complexes

The last analysis of the previous chapter focuses on protein complexes involving specific protein types, but regardless of the pairwise contacts within such a complex. This chapter refines these analysis for complexes whose atomic structure is known.

To see how, let a *template* of a complex be the graph whose nodes are the protein instances and the edges encode contacts between these instances. We compare such templates to the contacts observed between the toleranced proteins, at multiple scales. In doing so, we confirm / question / suggest protein contacts of the template based on the prominent contacts seen in the toleranced model,and hint at missing and or ill-placed proteins. Note that these tools can naturally be used to run in-silico experiments aiming at testing hypothesis.

1.5.5 Software

All the machinery described before requires elaborate implementations, in particular the CW VD and the λ-complex, as algebraic numbers are dealt with. The machinery also requires an advanced software design, as the interface should accommodate several implementations of the algorithms. To take a single example, a number of algorithms to compute toleranced models may be designed.

Chapter 7 presents our implementations, from two perspectives. The first one is of the *user manual* type, and presents the input, output and analysis available. The second one is of the *reference manual* type, and discusses precisely the C++ classes involved, and their traits parameters.

We note in passing that all the executable developed are made available from `http://cgal.inria.fr/abs/voratom`.

Chapter 2

Introduction (Version Française)

2.1 Reconstructions de gros systèmes par intégration de données

2.1.1 Machines macromoléculaires et fonctions biologiques

Les mécanismes biologiques reposant sur des complexes macromoléculaires, la compréhension de ces mécanismes d'un point de vue structural nécessite la description de ces complexes au niveau atomique. Dans sa forme la plus générale, cette tâche reste un problème ouvert posant de nombreuses questions. Premièrement, la durée de vie d'un complexe peut varier largement de quelques microsecondes pour un complexe transitoire (comme ceux impliquant les réactions d'oxydoréduction) à la durée de vie du système biologique le contenant (comme pour certaines enzymes avec plusieurs sous-unités). Comment pouvons nous caractériser la stabilité de ces complexes ? Deuxièmement, le nombre de partenaires d'une molécule peut varier dramatiquement. Quelle est la spécificité des interactions observées ?

De nombreux difficultés font obstacle à la résolution de ces questions, en particulier en ce qui concerne la taille des complexes étudiés, les plus gros assemblages moléculaires tel que les capsides virales ou le Complexe du Pore Nucléaire (NPC) étant composés de plusieurs centaines de chaînes polypeptidiques. De gros assemblages ont généralement une composition pouvant varier au cours du temps, posant des problèmes de plasticité (un assemblage particulier peut contenir différentes protéines à différents moments du cycle cellulaire). Par ailleurs, la conformation des molécules du complexe peut aussi changer au cours de la formation de l'assemblage, ou même pendant que l'assemblage est en activité (nous discuterons ce dernier cas dans le cadre du NPC), posant des problèmes de flexibilité.

Toutes ces difficultés motivent les activités de recherche à la frontière de la biophysique, de la biologie structurale et de l'informatique. Construire des modèles atomiques animés de ces machines moléculaires permettrait d'accroître notre compréhension de leur fonctionnement et des fonctions biologiques. Les expériences en laboratoire, fournissant des données à partir desquelles les modèles peuvent être développés et testés, sont une clé pour la modélisation. Pour des complexes de petite taille, des modèles atomiques peuvent être obtenus à partir d'analyses de cristallographie par rayons X ou de résonance magnétique nucléaire (RMN). Mais pour les raisons indiquees ci-dessus, il est beaucoup plus difficile de reconstruire de gros assemblages tels que les moteurs moléculaires (pour la mobilité des cellules), les filaments d'actine (pour la contraction musculaire), les protéines chaperonnes (pour le repliement des protéines) ou les complexes du pore nucléaire (pour la régulation du trafic nucléo-cytoplasmique)

2.1.2 Reconstruction des machines macromoléculaires par intégration de données

Les problèmes de modélisation rencontrés dans la reconstruction de gros assemblages sont différents de ceux inherents à l'amarrage protéique binaire, ou de ceux posés par la reconstruction de complexes de taille intermédiaire. Pour les complexes binaires, les problèmes principaux consistent actuellement à amarrer des molécules flexibles [LW10] et à créer des fonctions de score sachant identifier les complexes biologiques [FO10]. En particulier, l'expérience CAPRI (Critical Assessment of PRotein Interaction) invite les chercheurs à prédire en mode aveugle la structure de complexes dont la structure cristallographique a été résolue par ailleurs. Concernant

les complexes de taille intermédiaire, la reconstruction peut se faire en combinant le traitement d'images provenant de cryomicroscopie électronique (cryoEM) à des méthodes d'amarrage classique [LTSW09].
Pour les gros assemblages, ces méthodes ne permettent pas de restreindre suffisamment l'espace des solutions. Des stratégies plus sophistiquées doivent être employées, comme en particulier la reconstruction par intégration de données [AFK$^+$08]. Inspirée de la reconstruction à partir de données RMN, cette stratégie consiste à combiner des données expérimentales variées à fin de pouvoir cibler le ou les modèle(s) les plus cohérents avec ces données. Le paradigme actuel requiert trois ingrédients:

- Un modèle géométrique du système étudié. Pour les modèles gros grains, une famille de boules représentant les domaines des protéines est généralement utilisée.
- Des données expérimentales variées, éclairant différents aspects du système étudié. Ces données sont interprétées comme des *restrictions* (restraints en anglais), dont leur somme définissent une fonction de score mesurant la cohérence entre le modèle et les données.
- Une stratégie d'optimisation visant à trouver les minima les plus significatifs de la fonction de score.

Ces trois ingrédients peuvent alors être utilisés dans un processus mixant itérativement le calcul et la génération de données. Cette stratégie a été utilisée pour la reconstruction de modèles plausibles du NPC [ADV$^+$07a], voir Figure 2.1. Dans la suite, nous décrivons plus précisement ces trois ingrédients.

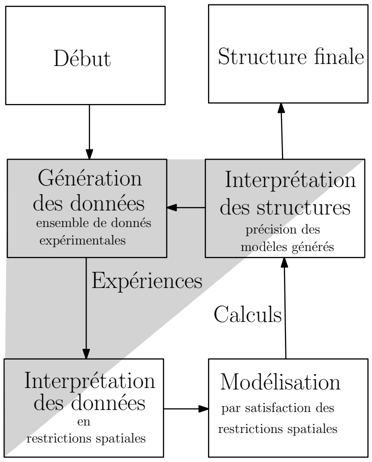


Figure 2.1: Les 4 étapes de la reconstruction par intégration de données, appliquées à la détermination de la structure globale du Complexe du Pore Nucléaire. Traduit à partir du matériel supplémentaire de [ADV$^+$07a].

2.1.3 Données biochimiques et biophysiques pertinentes, et leur difficultées inhérentes

Interpréter les données expérimentales en terme de restrictions qui serviront à contraindre le modèle à reconstruire n'est pas une tâche triviale. Avant de présenter ces restrictions dans le cas du NPC, nous discutons d'abord des données expérimentales les plus pertinentes dans ce cadre.

Méthodes TAP (Tandem Affinity Purification). Les méthodes TAP permettent d'accéder à tous les types de protéines trouvées dans tous les complexes contenant un type de protéines donné [PCR$^+$01], que nous appellerons *P*. Ces données peuvent donc être utilisées pour contraindre la proximité spatiale entre différents types de protéines dans un modèle.
Plus précisément, la génération de données TAP est constituée des étapes suivantes. D'abord, une protéine de fusion est créée en modifiant le gène de *P*: les séquences de deux marqueurs y sont intégrées, séparées par une séquence codant pour un site de clivage d'une protéase. Après avoir introduit le gène modifié dans une cellule hôte, la protéine modifiée (appelée protéine marqueur PrA, ou PrA-tagged protein) est exprimée et prend place dans au sein de ses complexes usuels — en supposant que les marqueurs ne fassent pas obstacle. En lysant la cellule, les complexes protéiques contenant les protéines de type *P* sont retrouvés à l'aide de deux étapes de purification. Chaque étape consiste à capturer ces complexes par purification sur colonne de chacun des

marqueurs. Entre les deux étapes de purification, les complexes retenus lors de la première étape sont libérés par une protéase coupant le premier marqueur au niveau du site de clivage. Ceci permet d'exposer le second marqueur pour la seconde étape de purification. Une fois ces étapes de purification faites, les complexes sont démantelés par électrophorèse, permettant d'obtenir un gel avec une bande pour chaque type de protéine dans le complexe. Les types présents sur le gel sont identifiés par spectrométrie de masse.

La liste des types de protéines ainsi obtenue (appelée en anglais pullout ou pulldown) appelle deux commentaires. Premièrement, il n'est pas possible de savoir si la liste des types de protéines correspond à la composition d'un ou plusieurs complexe(s). Par exemple, une liste (P,P',P'') obtenue en marquant P peut correspondre à un unique complexe contenant les trois types, ou bien à deux complexes différents contenant chacun (P,P') et (P,P''). Deuxièmement, la liste des types de protéines ne donne pas d'information sur la stoechiométrie des protéines au sein d'un complexe. Malgré les ambiguïtés combinatoires inhérentes aux données TAP, ces données sont d'intérêt primordial pour la reconstruction de gros assemblages: savoir que des instances de protéines participent à la formation d'un même complexe impose des restrictions sur les distances entre elles.

Essais de superposition (Overlay assays). Contrairement aux méthodes TAP, les essais de superposition ont pour but de détecter des contacts entre paires de protéines, permettant de contraindre sans ambiguïté les contacts entre protéines dans le modèle. Une protéine P_b appelée l'appât est en premier lieu purifiée et immobilisée sur un gel de nitrocellulose. Puis une protéine de fusion P_p appelée la sonde est à son tour purifiée et superposée avec P_b. Après une période d'incubation variable et une étape de lavage pour éliminer les sondes libres, les protéines superposées sont détectées [Hal04], fournissant un signal $S_{p,b}$ spécifique du complexe binaire sonde-appât.

Cependant, un bruit de fond perturbe le signal observé pour deux raisons particulières: premièrement, durant l'étape de purification, des particules non éliminées peuvent contaminer l'échantillon; deuxièmement, durant l'étape d'incubation, le nombre d'interactions non spécifiques entre appâts et sondes augmente linéairement avec le temps. Extraire un signal significatif devient alors un problème difficile.

Ultracentrifugation analytique. L'ultracentrifugation analytique permet de déterminer la forme globulaire (des domaines) des protéines, ce qui est particulièrement utile pour assigner à des protéines des formes ébauchées. Une centrifugeuse est une chambre réfrigérée, sous vide, contenant un rotor, alimenté par un moteur électrique, capable d'imprimer un grande vitesse de rotation. L'expérience consiste à faire tourner dans la centrifugeuse un échantillon de protéines. Deux forces principales opposées agissent: la force centrifuge et les forces de friction de l'échantillon avec le solvant. L'échantillon migre alors jusqu'à ce qu'il atteigne le fond ou le haut de son conteneur, ou bien jusqu'à ce que les forces en jeu atteignent un équilibre. En utilisant un solvant avec un gradient de densité (comme le sucrose), l'échantillon de protéines migre dans une zone du solvant partageant la même densité. En pratique, la densité des protéines est déterminée par comparaison de la migration de l'échantillon avec celles d'un ensemble de protéines marqueurs migrant dans la même zone du solvant. En mesurant la densité de l'échantillon, on détermine alors le coefficient de sédimentation S, qui est constant pour toutes les protéines marqueurs partageant la même densité que l'échantillon. Ce coefficient S est directement relié à la masse moléculaire, au volume et à la forme des protéines de l'échantillon [Eri09]. Intuitivement, pour des protéines de même volume, S est petit pour des protéines de forme allongée ayant une plus grand surface sensible aux forces de friction, alors que S est grand pour des protéines de forme globulaire.

Cryomicroscopie électronique (cryoEM). Une autre approche en plein essor est la cryomicroscopie électronique (cryoEM) [Fra06]. Des images des structures aussi grandes que des cellules, et aussi petites que des protéines, peuvent être obtenues en bombardant d'électrons des échantillons préalablement cryogénises, donnant accès dans le meilleur des cas à des images de résolution finale de l'ordre de 0.3 nanomètres. Deux modalités sont utilisées. Dans l'analyse de particules isolées (single particle analysis), le bombardement d'échantillons isolés permet d'obtenir des images correspondant à différents points de vue sur l'échantillon. Ces images peuvent être combinées en un modèle tridimensionnel. Dans la cryo-tomographie, un échantillon donné est bombardé sous différents angles, générant plusieurs images à partir desquelles un modèle tridimensionnelle peut aussi être reconstruit. Dans les deux cas, le modèle résultant est une carte de densité 3D où chaque voxel est muni d'une densité de la matière dans l'espace qu'il contient. En raison de la faible dose d'électrons requise pour ne pas abîmer les échantillons biologiques, cette densité est généralement très bruitée. Choisir un niveau de densité pour dessiner une surface (appelée l'enveloppe) enfermant le modèle est une tâche non triviale: les domaines globulaires des protéines correspondent à des zones de haute intensité alors que les régions non structurées, comme celles connectant les domaines globulaires, correspondent à des zones de faible intensité. Typiquement, la résolution obtenue en cryoEM varie de moyenne (autour de 5Å, permettant d'observer des éléments de structure secondaire), à faible (moins de 10Å, permettant d'observer des domaines). Dans les meilleurs cas, il est possible de plonger dans ces cartes des éléments structurales existants, ou résultants d'une modélisation, permettant ainsi de générer des modèles de résolution atomique.

Immunomicroscopie électronique (immunoEM). En immunoEM, le but est de localiser des protéines spécifiques au sein d'un assemblage [SH01], l'information sur leur positionnement permettant de favoriser la localisation des protéines dans le modèle. Pour y parvenir, des anticorps sont marqués avec des particules d'or puis, ces anticorps ciblant les antigènes d'intérêt. Les anticorps marqués sont ensuite examinés en microscope électronique. Les régions des images contenant des assemblages avec les particules d'or sont alors sélectionnées avec des cercles: le centre de chaque cercle est manuellement aligné avec l'assemblage. Après une étape de sélection de qualité, des fournées d'images d'assemblages sont manuellement alignées pour générer plusieurs montages; la position de toutes les particules d'or de chaque montage est alors mesurée.

L'impossibilité d'établir précisément les coordonnées des particules d'or est un problème majeur de l'immuno-localisation. De plus, pour plusieurs raisons (la rotation des anticorps autour des marqueurs, la distorsion ou les dégâts subits par les échantillons durant la préparation), chaque montage montre un haut degré d'éparpillement des particules d'or.

Ainsi et omission faite des essais de superposition, les données biochimiques et biophysiques ne peuvent être directement interprétées comme de simples restrictions spatiales. Par exemple, les listes de types de protéines obtenues avec les méthodes TAP ne révèlent directement aucune interaction binaire entre les types de protéines, introduisant un aspect combinatoire aux restrictions spatiales correspondantes. La variété des restrictions spatiales à inclure dans la fonction de score est aussi importante: il peut y avoir un biais dans le modèle final si les restrictions spatiales ne sont pas indépendantes (par exemple avec des restrictions spatiales sur les localisations et distances de plusieurs protéines), ou si une propriété particulière (comme la localisation des protéines) est représentée par différentes restrictions spatiales. Tous ces problèmes impliquent que les modèles calculés présentent des incertitudes qu'il est très difficile d'interpréter. Même si un modèle de haute qualité est produit par cette méthodologie, il est impossible de faire une évaluation précise sur la forme des protéines, leurs positions relatives et les contacts entre elles.

2.2 Le complexe du pore nucléaire: une description concise

Propriétés biologiques. Le Complexe du Pore Nucléaire (NPC) est le plus gros assemblage protéique de la cellule eucaryote connu à ce jour. Il est impliqué dans le trafic moléculaire à travers la membrane nucléaire, voire Figure 2.2, avec en particulier l'import des protéines ou l'export de l'ARN [WR10].

Le NPC est formé d'un canal de 100nm de diamètre, de filaments contenant des sites d'amarrage pour les molécules traversant le canal, et d'un panier du côté nucléaire. Les petites particules ($<$30kDa) peuvent passer à travers le NPC par diffusion passive, alors que les particules plus grandes doivent être reconnues par des filaments contenant des séquences spécifiques dites les séquences FG-répétées [DPU+03], aidant ainsi au transport actif d'un côté du pore à l'autre. Les protéines participant à ce processus actif sont appelées les karyophérines.

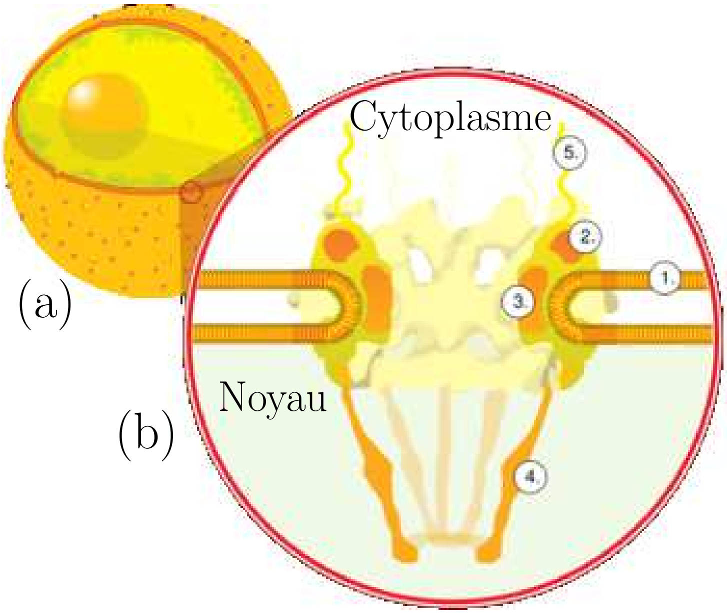


Figure 2.2: Structure grossière du Complexe du Pore Nucléaire (a) Les NPC sont localisés sur la membrane nucléaire. (b) Coupe d'une portion de membrane nucléaire: (1) La membrane nucléaire (2) L'anneau extérieur (3) Les rayons (4) Le panier (5) Les filaments. Traduit à partir de Wikipedia, `http://en.wikipedia.org/wiki/Nuclear_pore`.

Propriétés structurales. Les données expérimentales ont permis de montrer que le NPC est organisé en anneaux avec une symétrie d'ordre huit perpendiculairement au plan de la membrane nucléaire [ADV$^+$07b]. Il est en fait composé de 8 blocs identiques appelés les rayons, voire Figure 2.3. Chaque rayon se décompose en deux demi rayons, du côté cytoplasmique et du côté nucléaire.

Pour décrire les modèles existants du NPC, nous parlerons de *types* de protéines et d'*instances* de protéines, c'est à dire de copies d'un type donné. Dans un travail récent [ADV$^+$07b], le NPC a été modélisé avec 30 types différents, 29 instances de 27 types pour les demi rayons cytoplasmiques, et 28 instances de 25 types pour les demi rayons nucléaires. Les modèles du NPC comportent donc un total de $8 \times (29 + 28) = 456$ instances de protéines.

D'un point de vue global, le NPC peut être segmenté en quatre cylindres fonctionnels concentriques [HSBH07]: (i) le cylindre du canal qui contient les types avec des régions non structurées (les filaments impliqués dans la régulation du transport actif); (ii) le cylindre des adaptateurs, contenant les types intermédiaires entre les types de protéines du canal et ceux participant à l'architecture du NPC; (iii) le cylindre du manteau, contenant les types définissant l'architecture du NPC; (iv) le cylindre de la membrane du pore, contenant les types ancrant le NPC dans la membrane nucléaire.

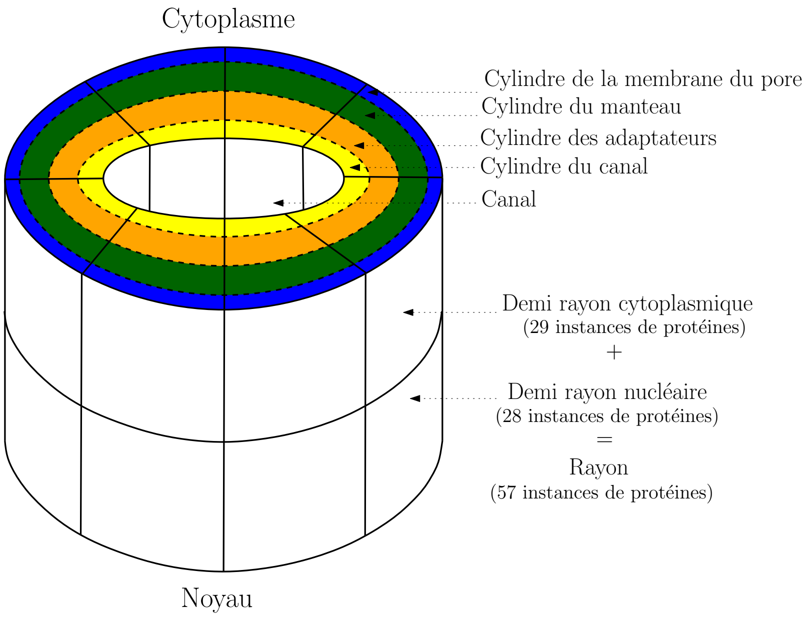


Figure 2.3: Représentation schématique de la structure du Complexe du Pore Nucléaire. Le NPC est composé de 8 rayons symétriques, chacun d'eux étant divisé en deux demi-rayons symétriques. Chaque rayon contient 57 protéines de 30 types différents localisés dans les cylindres du canal (jaune), des adaptateurs (orange), du manteau (vert foncé) ou de la membrane du pore (bleu). Adapté depuis [HSBH07].

2.3 Modélisation du complexe du pore nucléaire de la levure

Dans [ADV$^+$07a, ADV$^+$07b], Alber et al ont proposé un premier modèle gros grain du NPC de la levure basé sur l'intégration de données. Leur procédure de reconstruction leur a permis de sélectionner $N = 1000$ modèles plausibles. Nous allons maintenant brièvement récapituler les trois étapes de la procédure de reconstruction mentionnée dans la Section 2.1.2, puis discuter brièvement leur exploitation de ces N modèles.

2.3.1 Une hiérarchie de modèles gros grains

Une hiérarchie de modèles. Dans le modèle de [ADV$^+$07a], le NPC est représenté à l'aide d'une hiérarchie à 4 niveaux, voire Figure 2.4:

- (i) le niveau de l'assemblage, un modèle donné étant noté $A_i, i = \{1,\ldots,N\}$, où N est le nombre total de modèles;

- (ii) le niveau du demi rayon U_s^θ, où $s = 1,\ldots,8$ est l'indice du rayon, et l'indice $\theta \in \{1,2\}$ réfère respectivement au côté cytoplasmique ou nucléaire.

- (iii) le niveau de l'instance d'une protéine, une instance donnée étant notée $p_j, j = \{1,\ldots,456\}$.

- (iv) le niveau de la bille / particule, une bille B_k étant paramétrée par son centre et son rayon.

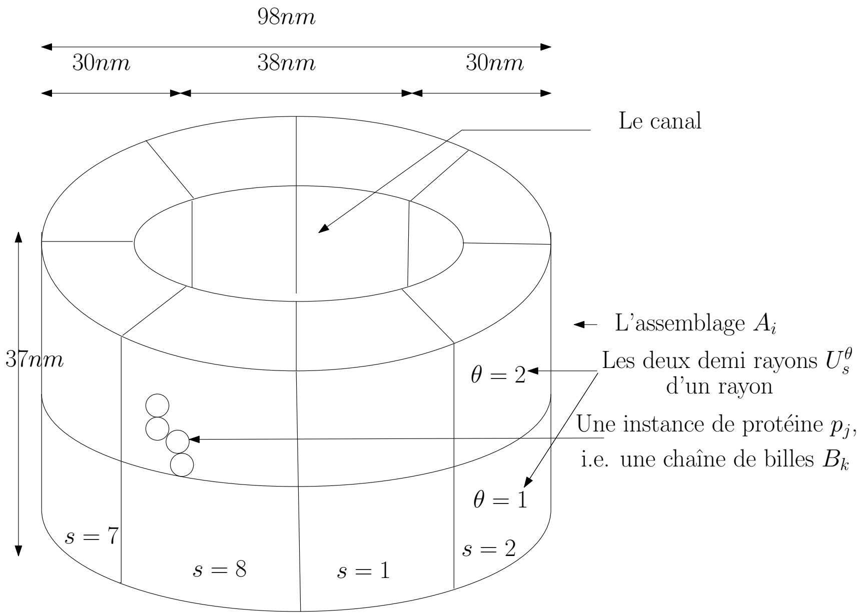


Figure 2.4: La représentation hiérarchique du Complexe du Pore Nucléaire: (i) l'assemblage A_i dans son ensemble (ii) le demi rayon U_s^θ (iii) l'instance de protéine p_j (iv) la bille B_k.

Avant la procédure d'optimisation, toutes les instances d'un type de protéine ont la même géométrie, consistant en une famille de billes de taille fixée. (La taille des billes dépend du type de protéine.) Le but de l'optimisation est de modifier les géométries initiales de manière à ce qu'elles deviennent cohérentes avec les données expérimentales.

Pour considérer toutes les interactions entre les instances de protéines dénuées de séquence FG-répétée à travers les différents rayons, les auteurs considèrent que chaque demi rayon a quatre demi rayons voisins, obtenus par la rotation de $2\pi/8$ des demi rayons cytoplasmiques. (Pour les protéines avec des séquences FG-répétées, les auteurs supposent que les filaments peuvent interagir avec des instance de protéines qui ne sont pas dans des demi rayons voisins.)

Finalement, la membrane nucléaire sert de moule pour l'assemblage qui s'y trouve ancré. La membrane nucléaire est représentée par des sphères de diamètre fixé correspondant à son épaisseur moyenne ($4.5nm$).

La représentation des instances de protéines. Dans le but de rendre compte de la diversité morphologique et biologique des différentes régions d'une protéine, les auteurs introduisent neuf niveaux de représentation numérotés de $k = 1$ à 9 – voire Figure 2.5. Le niveau $k = 1$ est la représentation racine, correspondant à la représentation la plus fine de la protéine. Toutes les autres représentations sont dérivées du premier niveau et permettent de considérer indépendemment les caractéristiques suivantes d'une protéine:

- niveau de représentation $k = 2$ et $k = 3$: domaines globulaires,
- niveau de représentation $k = 4$: régions non structurées,
- niveau de représentation $k = 5$: régions non transmembranaires,
- niveau de représentation $k = 6$: régions transmembranaires des protéines de la membrane du pore,
- niveau de représentation $k = 7$: régions non transmembranaires des protéines de la membrane du pore situées du côté périnucleaire,
- niveau de représentation $k = 8$: régions non transmembranaires des protéines de la membrane du pore situées du côté du pore,
- niveau de représentation $k = 9$: région C-terminal de Pom152, une protéine de la membrane du pore ayant des interactions homotypiques à travers sa région C-terminal dans le plan médian du NPC.

La représentation racine consiste en une chaîne orientée linéaire et flexible de billes identiques. Les billes sont choisies de manière à rentrer dans une ellipsoïde calculée à partir du coefficient de sédimentation de la protéine: la première bille représente la région C-terminal de la protéine alors que la dernière représente sa région N-terminal. Les coordonnées cartésiennes d'une bille dans les niveaux de représentations $k = 2$ à $k = 9$ ont les mêmes coordonnées qu'une des billes de la représentation racine, avec toutefois deux exceptions. Pour $k = 3$ et $k = 6$, une seule bille est utilisée, ses coordonnées étant calculées à partir d'une moyenne pondérée des coordonnées des billes aux niveaux de représentation respectifs $k = 2$ et $k = 1$.
⟶FREDERIC SAYS: PARAGRAPH CI-DESSUS: A REVISER⟵

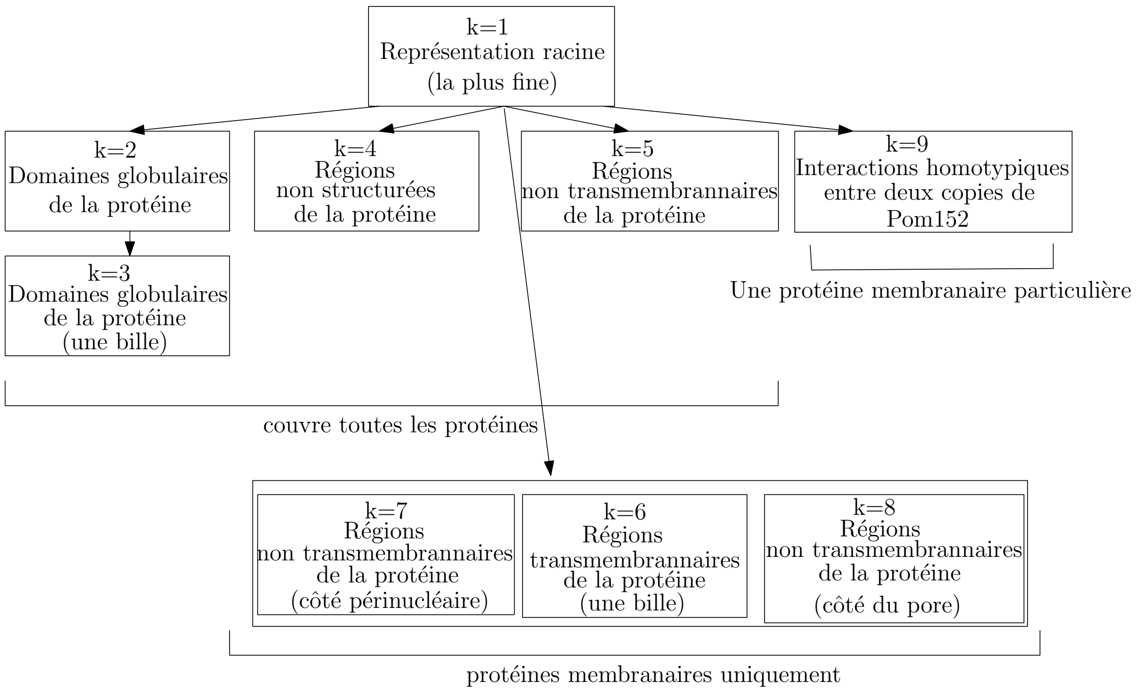


Figure 2.5: Les 9 niveaux de représentation des protéines dans le modèle de Alber et al. La flèche de $k = 2$ à $k = 3$ correspond à une vulgarisation de la représentation de la protéine. Toutes les autres flèches peuvent être vues comme des sélecteurs discriminant une région ou une caractéristique particulière de la protéine. On notera aussi que pour les niveaux de représentation $k = 3$ et $k = 6$, il n'y a forcément qu'une seule bille.

2.3.2 Les données et les restrictions associées concourant à la fonction de score

La fonction de score est interprétée comme une somme de restrictions spatiales, une restriction étant une fonction à valeur entière définie pour certains niveaux de représentations k, concourant à mesurer la cohérence du modèle avec les données expérimentales. La fonction associée à une restriction renvoie 0 si le modèle est cohérent avec les données, et une certaine valeur positive sinon. Il existe trois différents types de restrictions spatiales:

- les restrictions de localisation, contraignant la position des billes;
- les restrictions de distance, contraignant les distances entre deux billes;
- les restrictions de symétrie, contraignant deux ensembles de billes à avoir les mêmes propriétés;

Nous allons maintenant discuter ces restrictions comme une fonction des données disponibles.

La méthode TAP

Génération des données. Pour un type de protéine marqué, les auteurs obtiennent plusieurs listes de protéines, chacune qualifiee d'un facteur de qualité variant de 1 (haute qualité) à 3 (basse qualité). Les auteurs ont ainsi obtenu 75 listes contenant de 2 à 20 types de protéines, classées selon leur qualité: 34 de haute qualité, 21 de qualité intermédiaire et 20 de basse qualité.

⟶Frederic says: Restriction devient Contrainte?⟵

Restrictions. Les restrictions relatives aux données TAP sont utilisées pour les niveaux de représentation $2,4$ et 9. Rappelons qu'à cause de sa nature combinatoire, une liste de types de protéines ne peut être directement interprétée comme une restriction spatiale. Pour chaque demi rayon U_s^θ, une stratégie en deux temps est appliquée pour sélectionner les paires d'instances pour lesquelles un contact peut être contraint. Premièrement, les auteurs définissent un graphe complet pondéré dont les noeuds sont les types de protéines de la liste. Le poids d'une arête du graphe entre deux types P et P' est défini comme suit. Un score est calculé pour chaque paire d'instance de protéines p_i et p'_j de types P et P', de sorte qu'au moins l'une des deux instances soit dans U_s^θ , et l'autre dans le même demi rayon ou dans le voisinage. (Comme déjà mentionné, la sélection des instances de protéines pour les protéines avec des séquences FG-répétées n'est pas restreinte au voisinage.) Ce score est la distance minimale entre les centres des billes de p_i et p'_j. Alors, la paire d'instances de protéines minimisant le score est retenue et le poids de l'arête (P,P') devient ce score. Deuxièmement, un arbre couvrant minimal est calculé sur le graphe complet. Les arêtes sélectionnées définissent une connexion minimale des types de protéines. La somme des poids des arêtes sélectionnées forme le score de la restriction spatiale de la liste de types de protéines.

Les essais de superposition

Génération des données. Les auteurs ont générés 30 essais de superpositions correspondant aux sondes, i.e les 30 types de protéines incluses dans le modèle du NPC: ils ont obtenu ainsi une matrice de signaux $S_{p,b}$ correspondant à toutes les paires possibles des k sondes / n appâts. Pour réduire le rapport bruit /signal, les auteurs ont d'abord normalisé le signal $S_{p,b}$ de chaque appât (b) avec une sonde donnée (p) au signal général de fond de tous les appâts avec p. Une correction similaire a été appliquée pour le signal général de fond de chaque appât avec toutes les sondes pour générer un signal normalisé [1] pour chaque appât i avec chaque sonde j, $Y_{i,j}$:

$$Y_{i,j} = \frac{S_{i,j}}{\left(\sum_{i'=1}^{i'=n} S_{i',j} + \sum_{j'=1}^{j'=k} S_{i,j'} - S_{i,j}\right)} \tag{2.1}$$

Seule une interaction sonde / appât ayant un signal 10 fois supérieur à leur moyenne sont considérées comme significatives. En suivant le protocole, seulement 7 interactions binaires (faisant intervenir 5 types de protéine) sont qualifiées de significatives.

Restrictions. Les restrictions relatives aux essais de superposition sont utilisées pour les niveaux de représentation $2,4$ et 9. Les interactions binaires observées avec les essais de superposition sont contraintes en utilisant la même stratégie que pour les données TAP. Cependant, l'arbre couvrant minimal d'un graphe connexe à deux sommets étant lui même, l'aspect combinatoire disparaît.

L'ultracentrifugation analytique

⟶Frederic says: (a/b) ne veut rien dire: partie entiere superieure ou inferieure?⟵

⟶Frederic says: ci-dessous: derivent devient deduisent? de plus: inversion de quoi?⟵

[1] En fait, la formule fournie dans [ADV$^+$07a, Supplément, page 11] est:

$$Y_{i,j} = \frac{S_{i,j}}{\sum_{i'=1}^{i'=n} \sum_{j'=1}^{j'=k} S_{i',j'}}.$$

Toutefois, le signal est divisé par la somme de tous les signaux, ce qui est constant pour tous les signaux. Nous pensons que cette formule ne normalise pas les signaux.

Génération des données. Les auteurs utilisent l'ultracentrifugation analytique pour déterminer les billes à la représentation racine d'une protéine. Chaque protéine est représentée par une ellipsoïde de révolution (i.e une ellipsoïde obtenue par rotation d'une ellipse autour de son axe majeur). Le nombre de billes par protéine est l'entier le plus proche du rapport axial (a/b) de l'ellipsoïde, et le rayon des billes est choisi de manière à reproduire au mieux le volume de la protéine, lequel est estimé à partir de sa séquence d'acides aminés. Pour calculer le rapport axial, les auteurs calculent la fonction de Perrin P (une fonction de forme moléculaire) du coefficient de sédimentation S du type de la protéine. En utilisant la formule d'inversion d'un ellipsoïde de révolution avec un développement d'ordre n (n est généralement égal à 6) [HC95], on peut obtenir la formule suivante:

$$(a/b) = a_0 + \sum_{i=1}^{i=n} a_i P^i \tag{2.2}$$

où a_0 et $(a_i)_{i \leq n}$ sont des paramètres donnés [HC95, Table 3 et 4].

Les auteurs utilisent aussi l'ultracentrifugation analytique pour estimer la longueur D de l'axe le plus long de l'ellipsoïde contenant le complexe Y, un complexe protéique bien connu composé de 7 types de protéine – voir les détails dans la Section 4.2.1.

Restrictions. Trois restrictions spatiales dépendent uniquement de l'ultracentrifugation analytique: la première contraint n'importe quelle paire de billes à avoir une intersection vide; la deuxième contraint deux billes consécutives dans une instance de protéine à être tangentes; la dernière contraint les billes des instances de protéine du complexe Y dans un demi rayon du NPC à être contenues dans une sphère de diamètre D.

La cryomicroscopie électronique

Génération des données. La carte obtenue à partir de l'analyse de particules isolées a été utilisée dans l'établissement de plusieurs caractéristiques globales importantes: les symétries d'ordre huit (huit rayons) et d'ordre deux (deux demi rayons par rayon), l'ancrage du NPC dans la membrane nucléaire et le diamètre du canal. La carte de densité a aussi été utilisée comme référence pour la génération des cartes d'immunomicroscopie électronique.

Restrictions. Les données cryoEM sont utilisées pour deux types de restrictions spatiales. Premièrement, une restriction de distance pour le niveau de représentation 5 contraint les billes des instances de protéine et les billes de la membrane nucléaire à s'exclure mutuellement. Deuxièmement, à cause des symétries d'ordre huit et deux, deux restrictions de symétrie pour le niveau de représentation 2 contraignent deux instances de protéine (p_i, p_j) d'un type P présentes dans deux différents demi rayons à partager la même configuration de billes: (i) d'abord en contraignant deux paires de billes de p_i et p_j à avoir une distance identique, (ii) ensuite en contraignant deux ensembles de quatre billes de différentes instances de protéine dans deux demi rayons différents à partager le même angle diédral.

L'immunomicroscopie électronique

Génération des données. Exceptées deux protéines (Nup1 et Nup60), les auteurs ont collecté 300 positions de particules pour chaque type de protéine, en alignant les cercles dessinés sur les images de microscopie électronique avec l'intersection entre l'axe Z central et le plan équatorial du NPC. Pour prendre en compte l'éparpillement inhérent, les auteurs ont créé un scénario dans lequel la position des protéines était connue – la distribution du modèle. Plus précisément, une membrane plastique a été recouverte de protéines marquées, puis observée de manière à ajuster une distribution de Gausse recouvrant la distribution du modèle. Cette distribution de Gausse a ensuite été glissée le long des axes Z et R de manière à maximiser le chevauchement avec la distribution expérimentale des particules d'or observées. Le chevauchement est défini comme suit:

$$Chevauchement = 1 - \frac{1}{2} \sum_{i=1}^{N_b} |E_i - C_i| \tag{2.3}$$

où E et C sont respectivement les distributions expérimentales et calculées normalisées, et N_b est le nombre de boîtes de dimension 2.5nm couvrant les distributions de 0 à une valeur positive suffisamment grande.

Finalement, les auteurs corrigent les erreurs de localisation dues à des gênes stériques entre la membrane nucléaire et le NPC. Ces portions de structure sont plongées dans la structure du NPC obtenue par la carte de densité de référence de cryoEM. Au final, la totalité des cartes de immunomicroscopie électronique est alignée de manière à maximiser le chevauchement avec la carte de référence.

Restrictions. Les restrictions relatives à l'immmunomicroscopie électronique sont utilisées pour le niveau de représentations 1. Deux restrictions de localisation sont définies pour favoriser les coordonnées des axes Z et R de toutes les billes des instances de protéine (exceptées Nup1 et Nup60) à être dans un intervalle donné.

2.3.3 Les méthodes d'optimisation

Comme dit dans la Section 2.3.1, une protéine est représentée par une collection de billes. Dans la suite, nous appelons *configuration* du NPC la position et la taille de toutes les billes à la représentation racine. On notera que si les rayons de toutes les billes peuvent être estimés de manière à atteindre les volumes estimés des protéines à partir de leur séquence d'acides aminés, alors la dimension de l'espace des configurations est exactement trois fois le nombre de billes.

Stratégie générale. La stratégie d'optimisation se fait en deux étapes, comme illustrer sur la Figure 2.6. La première étape dite gros grain permet de sélectionner 200 000 configurations. Puis la seconde dite fin grain, démarrant avec les 20 000 configurations ayant le meilleur score parmi les 200 000 précédentes, raffine cette sous sélection. Finalement, les 1000 configurations avec le meilleur score sont retenues. L'optimisation d'une configuration A_i du NPC est réalisée en minimisant une fonction de score $f(A_i)$ consistant en une combinaison linéaire d'un sous ensemble des restrictions spatiales. Toutes les restrictions ne sont pas ajoutées en une seule fois à la fonction de score, mais sont à la place graduellement ajoutées au cours de l'optimisation [BG85]. La minimisation de $f(A_i)$ est réalisée par une conjugaison de descente de gradients et de dynamique moléculaire couplées avec du recuit simulé. Pour chaque configuration A_i, dix de ces conjugaisons sont réalisées durant l'étape gros grain, et au plus huit durant l'étape fin grain.

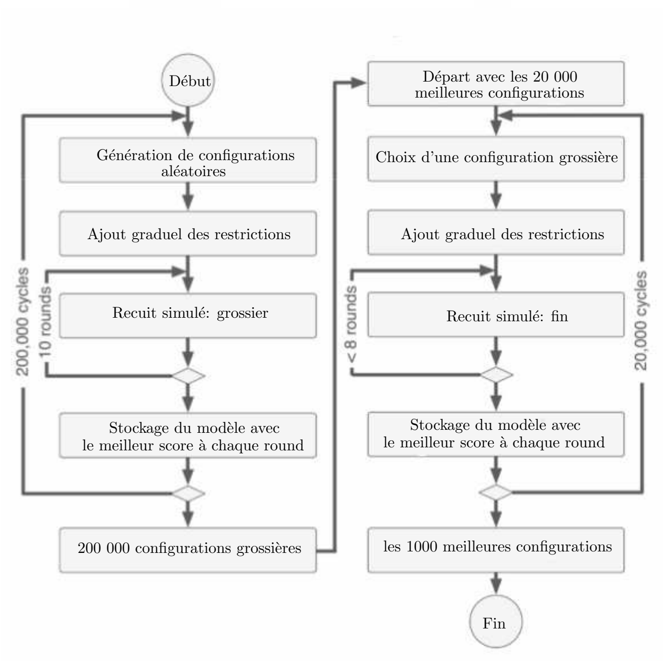

Début
Génération de configurations aléatoires
Ajout graduel des restrictions
Recuit simulé: grossier
Stockage du modèle avec le meilleur score à chaque round
200 000 configurations grossières
200,000 cycles
10 rounds
Départ avec les 20 000 meilleures configurations
Choix d'une configuration grossière
Ajout graduel des restrictions
Recuit simulé: fin
Stockage du modèle avec le meilleur score à chaque round
les 1000 meilleures configurations
Fin
< 8 rounds
20,000 cycles

Figure 2.6: Les deux principales étapes du protocole d'optimisation: l'optimisation gros grain (à gauche), et l'optimisation fin grain (à droite). Traduit de la Figure 11 [ADV$^+$07a, Supplément].

2.3.4 Les résultats et leur évaluation

Les $N = 1000$ structures ne sont pas accessibles, mais deux produits dérivés le sont: les fréquences de contact, et les cartes de densité de probabilité. Ces résultats sont disponibles sur `http://salilab.org/npc/`. Nous allons maintenant discuter brièvement ces résultats.

Signifiance des fréquences de contact. Les *fréquences de contact* d'une paire de types de protéine (P, P') est la fraction de modèles optimisés, parmi les $N = 1000$, ayant au moins un contact entre une instance p de P et une instance p' de P'. Il y a un contact entre deux instances si il y a au moins une paire de billes, une de p et une de p', tel que la distance entre leur centre satisfasse une contrainte de distance. Notons que si deux instances particulières de protéines de types P et P' sont en contact dans tout les modèles, mais que toutes les autres paires possibles d'instances de ces deux types ne sont pas en contact, la fréquence de contact est égal à 1. Ceci montre que les fréquences de contact ne prennent pas en compte la stoechiométrie des contacts.

Incertitudes dans les cartes de densité de probabilité. Pour exploiter globalement les $N = 1000$ modèles sélectionnés, les auteurs ont calculé une *carte de densité de probabilité* par type de protéine. Plus précisément, considérons un type de protéine particulier. Après avoir collecté toutes les billes des instances de protéine de ce type dans les N modèles, les auteurs mélangent ces billes à fin de produire une carte de densité de probabilité représentant la position moyenne de toutes ces instances.

Comme représenter dans la Figure 2.7, choisir un niveau de densité dans ces cartes permet d'avoir une représentation des protéines. Ces cartes de densité offre un résultat qualitatif sur les formes et les positions des protéines au sein du NPC. Cependant, due à l'incertitude inhérente aux données biochimiques et biophysiques, et au biais introduit par la méthodologie, il est très difficile de faire une analyse quantitative de ces cartes. Pour exemple extrême, l'anneau autour du NPC dans la Figure 2.7 correspond à la carte de densité de 16 instances de Pom152.
Les citations suivantes témoignent de la nature qualitative des analyses menées par [ADV$^+$07a, ADV$^+$07b]:
$\ll$ *Our map is sufficient to determine the relative positions of proteins in the NPC; we do not interpret features smaller than this precision.* $\gg$
(Nos cartes sont suffisantes pour déterminer les positions relatives des protéines du NPC; nous n'interprétons pas les caractéristiques plus petites que cette précision là)
$\ll$ *Because of the limited precision of the information used here, the localization volume of a protein should not be mistaken for its density map, such as that derived by cryo-EM.* $\gg$
(À cause de la précision limitée de l'information utilisée ici, le volume de localisation d'une protéine ne doit pas être confondu avec sa carte de densité, comme celle provenant de cryoEM)
$\ll$ *The localization volumes [...] allow a visual interpretation of the relative proximities of the proteins.* $\gg$
(Le volume de localisation [...] permet une interprétation visuelle de la proximité relative des protéines)

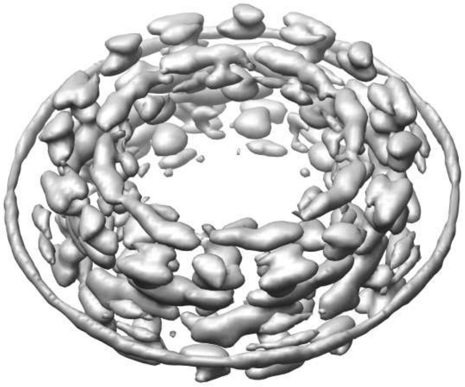


Figure 2.7: Une surface de niveau correspondant a un seuil de 0.5 dans une carte de densité de probabilité 3D résultante d'une reconstruction du NPC. Le logiciel CHIMERA a été utilisé pour l'affichage.

2.4 Les diagrammes de Voronoï et les α-complexes

2.4.1 Traitement des incertitudes avec les diagrammes de Voronoï affines et leur α-complexe

Diagrammes de Voronoï, partitions de l'espace, et modèles paramètres. Comme nous l'avons vu en présentant les reconstructions du NPC, les incertitudes sur les entrées de la procédure sont directement traduites sur les sorties. En particulier, étant donnée une carte de densité de probabilité, placer les instances des protéines est un problème mal posé, puisqu'en général, il n'y a pas d'unique solution. D'un autre coté, dans un certain nombre de cas, il est possible de délimiter une zone dans laquelle les phénomènes intéressants peuvent être confinés. Cette zone peut être alors utilisée comme support d'un modèle paramétré balayant la région d'intérêt. Naturellement, confiner le modèle dans une région peut revenir à chercher les caractéristiques de ce modèle. Cet argument est particulièrement vrai pour les cartes de densité de probabilité si les incertitudes dans ces cartes sont identiques aux incertitudes dans les modèles paramétrés.

Ce dernier paradigme semble attractif, mais le problème clé est de définir des formes paramétrées. À cette fin, un choix de design consiste à considérer une paramétrisation revenant à des formes emboîtées. Le cas classique en géométrie algorithmique est l'α-forme pondérée associée à un diagramme de puissance, une construction remarquable proposée par E. Edelsbrunner [Ede92]. Pour un soucis de clarté, nous allons présenter le cas plus simple des α-formes associées aux diagrammes de Voronoï Euclidiens.

Diagrammes de Voronoï Euclidiens d'un ensemble de points. Formellement, étant donné un ensemble $\mathscr{S}$ de n points dans l'espace Euclidien $\mathbb{E}$, le diagramme de Voronoï Euclidien de $\mathscr{S}$ est la partition de l'espace en *cellules de Voronoï* telle que chaque cellule de Voronoï contienne tous les points de l'espace plus proche d'un point $x_i \in \mathscr{S}$ que n'importe quel autre point $x_{j \neq i} \in \mathscr{S}$ par rapport à la distance Euclidienne:

$$Vor(x_i) = \{x \in \mathbb{E} \text{ tel que } \mid\mid x - x_i \mid\mid \leq \mid\mid x - x_j \mid\mid, \forall x_j \neq x_i\}. \tag{2.4}$$

En 2D, tous les points à égale distance de deux autres points $x_i \in \mathscr{S}$ et $x_j \in \mathscr{S}$ sont localisés sur une droite appelée le *bissecteur*. L'intersection des cellules de Voronoï de x_i et x_j est une portion connexe du bissecteur appelée *arête de Voronoï*. Deux arêtes de Voronoï (ou génériquement trois cellules de Voronoï) s'intersectent en un point appelée le *sommet de Voronoï*. Soit v un sommet de Voronoï. Puisque v est à égale distance de trois points $(x_i, x_j, x_k) \in \mathscr{S}$, il est le centre d'un cercle B circonscrite au triangle (x_i, x_j, x_k). De plus, puisque v est plus proche de x_i, x_j, x_k que n'importe quel autre point de $\mathscr{S}$, le cercle B ne contient aucun autre point de $\mathscr{S}$ dans son intérieur. Cette propriété est appelée la propriété de *la boule vide* ou encore la propriété *sans conflit* et peut être étendue aux points d'une arête de Voronoï ou d'une cellule de Voronoï.

Structure duale d'un diagramme de Voronoï Euclidien. Le diagramme de Voronoï a une structure duale appelée la triangulation de Delaunay, laquelle est obtenue en collectant les sommets, arêtes et triangles comme suit:
– une cellule de Voronoï associée à un point x_i correspond à un sommet de Delaunay;
– une arête de Voronoï $e_{i,j}$ associée à deux points x_i et x_j correspond à une arête de Delaunay (x_i, x_j), obtenue en joignant x_i et x_j;
– un sommet de Voronoï $v_{i,j,k}$ associé à trois points x_i, x_j et x_k correspond à un triangle de Delaunay (x_i, x_j, x_k), lequel peut être vu comme l'enveloppe convexe des trois points.

Génériquement, si il n'y a pas quatre points cocirculaires en 2D, cette structure duale est une triangulation. À cause de la propriété de la boule vide, les cercles circonscrits aux triangles de Delaunay n'ont aucun point de $\mathscr{S}$ dans leur intérieur. De la même manière, pour chaque point contenu dans l'intérieur d'une arête de Voronoï, il existe un cercle sans conflit passant par les sommets de l'arête de Delaunay duale. Nous notons aussi que le cercle circonscrit $B_{i,j}$ d'une arête duale (x_i, x_j), i.e le cercle dont le diamètre est (x_i, x_j), est sans conflit si et seulement si son centre est localisé sur $e_{i,j}$. Une arête duale ayant cette propriété est dite de *Gabriel*.
La triangulation est en fait un complexe simplicial, c'est à dire une famille de simplexes tel que pour deux simplexes qui s'intersectent, ils s'intersectent sur une face commune. Avec cette terminologie, la triangulation de Delaunay consiste en une famille de k-simplexes, où k est la dimension du simplexe: un sommet de Delaunay est un 0-simplexe; un arête de Delaunay est un 1-simplexe; un triangle de Delaunay est un 2-simplexe. Le dual d'un

k-simplexe est une cellule de Voronoï ($k = 0$), une arête de Voronoï ($k = 1$) ou un sommet de Voronoï ($k = 2$): nous appelons $2 - k$-face le dual d'un k-simplexe.

Diagramme de remplissage de l'espace, l'α-complexe et l'α-forme. Le diagramme de Voronoï admet la construction intuitive suivante: faisons grandir des sphères centrées sur les points de $\mathscr{S}$, à la même vitesse [2] $\sqrt{\alpha}$: les sphères s'intersectent sur leur bissecteur commun. Le domaine couvert par les boules bordées par ces sphères grandissantes est appelé le diagramme de remplissage de l'espace. Nous appelons la *restriction d'une boule* son intersection avec sa région de Voronoï. La construction précédente revient à suivre les intersections entre restrictions. Notons que cette stratégie est plus restrictive que de suivre simplement l'intersection entre les sphères, comme l'illustre la croissance de trois sphères formant un triangle obtus, voire Figure 2.8.

De la même manière pour la construction de la triangulation de Delaunay, il est possible de trouver les simplexes correspondant à l'intersection des restrictions. L'ensemble de ces simplexes forment un autre complexe simplicial appelé l'α-complexe, dont le domaine (l'union du domaine géométrique des sommets, arêtes et triangles) définit l'α-forme.

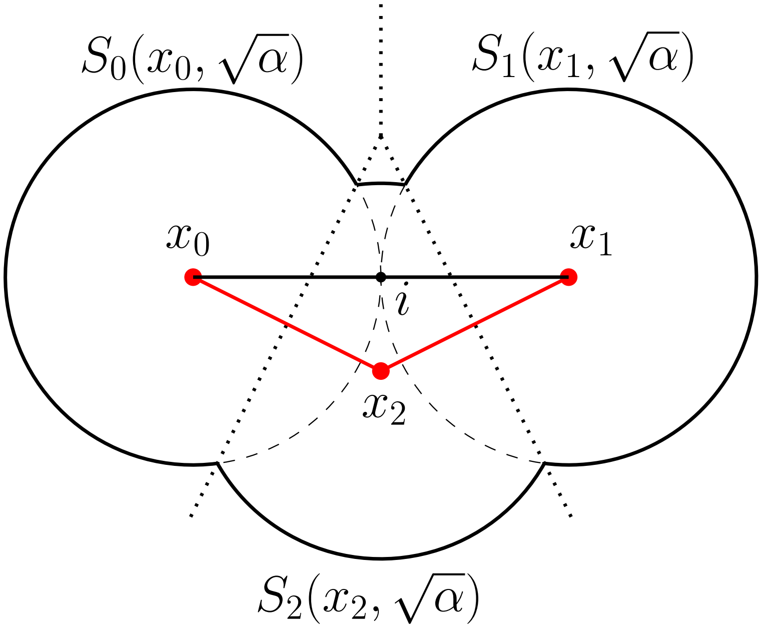

$S_0(x_0, \sqrt{\alpha})$
$S_1(x_1, \sqrt{\alpha})$
x_0
x_1
i
x_2
$S_2(x_2, \sqrt{\alpha})$

Figure 2.8: Un exemple 2D de l'α-complexe de trois points x_0, x_1 et x_2, pour $\alpha > 0$. Le diagramme de Voronoï est dessiné en pointillés, et la triangulation de Delaunay en lignes solides. Les points rouges et les lignes rouges correspondent aux simplexes dans l'α-complexe. Les restrictions des sphères $S_0(x_0, \sqrt{\alpha}), S_1(x_1, \sqrt{\alpha})$ et $S_2(x_2, \sqrt{\alpha})$ sont montrées en lignes noires solides. S_1 et S_2 sont dessinées en traits interrompus. Notons que S_0 et S_1 s'intersectent au point i à l'intérieur de la boule bordée par S_2: le simplexe dual correspondant n'est pas de Gabriel.

Notons que faire croître les sphères est un processus monotone: quand les rayons grandissent, les restrictions s'emboîtent, et il en va de même pour les α-complexes. En fait, le 0-complexe correspond à l'ensemble des points de $\mathscr{S}$, et pour une valeur suffisamment grande, l'α-complexe correspond à la triangulation de Delaunay entière. Notons aussi que le processus de croissance monotone précédent correspond à l'emboîtement discuté dans la Section 2.4.1. En fait, la contribution méthodologique de cette thèse repose sur une généralisation de l'α-complexe. Pour positionner cette contribution, nous allons maintenant brièvement présenter trois diagrammes de Voronoï classiques.

[2] le choix de la vitesse $\sqrt{\alpha}$ plutôt que α sera éclairci lorsque nous discuterons des diagrammes de puissance.

2.4.2 Les diagrammes de Voronoï généralisés

Un diagramme de Voronoï étant une partition de l'espace ambiant induit par une distance (généralisée) entre objets, de nombreux diagrammes de cette espèce peuvent être définis. Trois d'entre eux sont particulièrement intéressants, et leurs spécifications sont récapitulées dans la Table 2.1. Dans tous les cas, la distance généralisée est de la forme:

$$a_i \mid\mid x - c_i \mid\mid^k - b_i, \tag{2.5}$$

où les paramètres a_i et b_i sont respectivement appelés les paramètres multiplicatif et additif , et $k \in \mathbb{N}^*$ est la puissance de la distance. Nous référons le lecteur à [BWY06] pour une description plus précise de ces diagrammes.

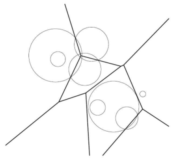
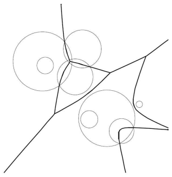
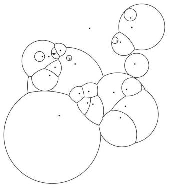
Nom	Distance généralisée	Bissecteur	Diagramme en 2D
Diagramme de puissance de sphères: $S_i(c_i, w_i = r_i^2)$	$d(S_i(c_i, w_i), x)$ $=$ $\mid\mid c_i - x \mid\mid^2 - w_i$	Hyperplans radicaux	
Diagramme d'Apollonius de sphères: $S_i(c_i, r_i)$	$d(S_i(c_i, r_i), x)$ $=$ $\mid\mid c_i - x \mid\mid - r_i$	Hyperboloïdes ou droites	
Diagramme de Möbius de points additif-multiplicatifs: $W_i(c_i, a_i, b_i)$	$d(W_i(c_i, a_i, b_i), x)$ $=$ $a_i \mid\mid c_i - x \mid\mid - b_i$	Hypersphère (possiblement dégénérées)	

Table 2.1: Les diagrammes de Voronoï classiques.

Le diagramme de puissance. Ce diagramme est défini pour les sphères et généralise la distance à la puissance entre sphères. Dans la suite, la dimension de l'espace ambiant est notée d, et les objets considérés sont des sphères. Ce diagramme généralise le cas Euclidien, puisque tous les diagrammes de Voronoï affines sont des diagrammes de puissance. Le bissecteur de deux sphères est appelé l'hyperplan radical. Tout point situé sur une $d-k$-face de Voronoï est le centre d'une sphère qui est orthogonale à $k+1$ sphères de $\mathscr{S}$ et sur-orthogonale à toutes les autres. [3]. Deux situations intéressantes non rencontrées dans le cas Euclidien peuvent apparaître: une cellule de Voronoï d'une sphère $S_i \in \mathscr{S}$ (i) peut ne pas contenir le centre de S_i (le duale de la cellule de Voronoï n'est pas de Gabriel) ou (ii) peut être vide (S_i est *cachée*).

[3] Deux sphères $S_i(c_i, r_i)$ et $S_j(c_j, r_j)$ sont orthogonales si et seulement si $\mid\mid c_i - c_j \mid\mid^2 = r_i^2 + r_j^2$.

Le diagramme d'Apollonius. Il s'agit d'une autre généralisation du cas Euclidien aux sphères, en utilisant cette fois ci une distance additive. (La distance additive réfère à la distance d'un point à une sphère, plutôt qu'à son centre. Il s'en suit que le rayon agit comme un paramètre additif.) Les points sur le bissecteur de k sphères dans $\mathscr{S}$ sont les centres de sphères tangentes à ces k sphères. En 2D, le bissecteur peut être une branche d'une hyperbole ou bien une ligne droite. Une cellule de Voronoï d'une sphère $S_i \in \mathscr{S}$ (i) contient toujours le centre de S_i (son dual est toujours de Gabriel) et (ii) est vide si et seulement si il existe une autre sphère $S_j \in \mathscr{S}$ contenant S_i. Notons que deux sphères $(S_i, S_j) \in \mathscr{S}$ peuvent avoir plusieurs arêtes de Voronoï en commun, et que trois sphères $(S_i, S_j, S_k) \in \mathscr{S}$ peuvent avoir au plus deux sommets de Voronoï en commun.

Le diagramme de Möbius. Il s'agit d'une généralisation du diagramme de puissance aux points additif-multiplicatifs, et à la puissance additive-multiplicative. Pour deux points additif-multiplicatifs, le bissecteur est une hypersphère ou une ligne droite. Une cellule de Voronoï d'un point additif-multiplicatif $W_i \in \mathscr{S}$ (i) n'a pas nécessairement de simplexe dual de Gabriel et (ii) peut être vide. Notons que un point additif-multiplicatif $(W_i) \in \mathscr{S}$ peut avoir plusieurs cellules de Voronoï, que deux points $(W_i, W_j) \in \mathscr{S}$ peuvent avoir plusieurs arêtes de Voronoï en commun et que trois points $(W_i, W_j, W_k) \in \mathscr{S}$ peuvent avoir au plus deux sommets de Voronoï en commun.

2.5 Aperçu de la thèse

D'un point de vue pratique, cette thèse a pour but de corriger les limitations des modèles mentionnés dans la Section 2.3.4. D'un point de vue méthodologique, ce but est motivé par le développement d'algorithmes et de concepts relatifs aux diagrammes de Voronoï courbes. Nous allons maintenant introduire ces contributions.

2.5.1 Les diagrammes de Voronoï additif-multiplicatif et les λ-complexes

Dans le but de modéliser des assemblages macromoléculaires avec de l' incertitude, nous introduisons les *modèles tolérancés* pour accommoder les incertitudes sur le positionnement et la conformation des instances de protéine au sein de ces assemblages. Un modèle tolérancé est une famille de *boules tolérancées*, chacune de ces boules consistant en deux boules concentriques appelées les boules *intérieure* et *extérieure*, encodant respectivement les régions de haute et faible confiance. L'interpolation linéaire des rayons des boules tolérancées définit un processus de croissance équivalent au *diagramme de Voronoï additif-multiplicatif* (CW).

Dans le Chapitre 3, nous introduisons en premier lieu les modèles tolérancés. Puis nous généralisons la propriété de la boule vide, et examinons les propriétés des bissecteurs du CW, du diagramme lui-même, et de sa structure duale. Puis nous procédons à une généralisation de l'α-complexe que nous appelons le λ-complexe. Finalement, nous présentons un algorithme naïf sensible à la complexité de la sortie pour calculer une représentation abstraite du dual du CW et de son λ-complexe.

2.5.2 Le complexe du pore nucléaire: matériel et méthodes

Le complexe du pore nucléaire est le plus grand assemblage macromoléculaire connu dans la cellule eucaryote, et comprendre sa structure et sa fonction est un problème clé de la biologie cellulaire et structurale. En particulier, analyser la position et les partenaires de chaque instance de protéine est nécessaire pour la détermination de la structure de l'assemblage. Le modèle du NPC de Alber et al présenté en Section 2.3 est une première étape, mais les résultats de nature qualitative font de l'interprétation du modèle une tâche très difficile.

Le Chapitre 4 introduit les caractéristiques clés du NPC. Nous introduisons d'abord trois importants sous-complexes du NPC (le complexe Y, le complexe T et le complexe Nup82), jouant des rôles clés d'un point de vue structural et fonctionnel. Nous présentons ensuite une brève évaluation des cartes de densité de probabilité de [ADV$^+$07a]. Enfin, nous présentons des algorithmes pour construire le modèle tolérancé du NPC basé sur ces cartes de densité de probabilité.

2.5.3 Évaluation des contacts binaires aux niveaux de l'assemblage et des sous-complexes

Le Chapitre 5 présente un certain nombre d'analyses sur le NPC, basées sur le modèle tolérancé présenté dans le chapitre précédent, et sur le λ-complexe de ce modèle. Le but général est de présenter une analyse multi-échelle des contacts entre types de protéine, mais aussi de faire une investigation sur des complexes protéiques du NPC faisant intervenir certains types de protéine.

D'abord, nous présentons le *diagramme de Hasse* d'un modèle tolérancé, qui est l'interprétation du λ-complexe en terme d'instances de protéine. Nous utilisons l'historique des contacts entre protéines contenu dans le diagramme de Hasse pour définir la courbe de contacts de deux types de protéine, une courbe contenant une information dépendant de la stoechiométrie des contacts entre instances de protéine de ces deux types.

Finalement, nous nous concentrons sur des complexes faisant intervenir des types de protéine spécifiques, en présentant une analyse visant à évaluer les propriétés de symétrie du modèle tolérancé. Cette tâche particulière est réalisée dans le cadre des trois sous-complexes décrits dans le Chapitre 4.

2.5.4 Évaluation de modèles graphiques de sous-complexes

La dernière analyse du chapitre précédent se concentre sur des complexes protéiques impliquant des types de protéine spécifiques, mais sans se préoccuper des contacts binaires au sein du complexe. Le Chapitre 6 raffine cette analyse pour des complexes de structure atomique connue, ou pourvus d'un modèle de résolution atomique. Pour voire comment, nous définissons le *patron* d'un complexe comme le graphe dont les noeuds sont les instances de protéine et les arêtes encodent les contacts entre ces instances. Nous comparons de tels patrons aux contacts observés entre les protéines du modèle tolérancé, sur plusieurs échelles. Ainsi, nous pouvons confirmer, questionner ou suggérer des contacts entre protéines du patron en se basant sur les contacts les plus observés dans le modèle tolérancé, et cibler les protéines manquantes ou mal positionnées. Notons que ces outils peuvent naturellement être utilisés pour des expériences in silico visant à tester des hypothèses.

2.5.5 Logiciel

Toute la machinerie décrite ci-dessus requiert une implémentation élaborée, en particulier pour le CW et son λ-complexe demandant des calculs potentiellement compliqués sur des nombres algébriques. La machinerie requiert aussi un design de logiciel avancé, de manière à ce que l'interface puisse accommoder différents algorithmes. Pour prendre un exemple simple, différents algorithmes pour le calcul d'un modèle tolérancé peuvent être envisagés.

Le Chapitre 7 présente nos implémentations, sur deux points de vue. Le premier est de type *manuel d'utilisateur*, et présente les entrées, les sorties et les analyses disponibles. Le second est de type *manuel de référence*, et discute précisément les classes C++ impliquées, et comment les paramétrer.

Nous notons au passage que tous les exécutables développés sont accessibles via `http://cgal.inria.fr/abs/voratom`.

Chapter 3

Compoundly Weighted Voronoi Diagrams and their λ-Complex

3.1 Introduction - Rationale

Spatial partitions have a long standing history in science and engineering, as they allow allocating regions to specific objects, a general framework to model growth processes. The classical spatial partition is the usual Euclidean Voronoi diagram, which is a particular case of the power diagram, the most general affine Voronoi diagram. In a power diagram, balls are grown by adding a quantity α to their squared radii.
In this chapter, we develop the Voronoi diagram associated with a growth process which consists of linearly interpolating the radii of a collection of balls. To begin with, we show in section 3.2 that this growth process is equivalent to a so-called compoundly weighted (CW) Voronoi diagram. In Section 3.3, we proceed with the analysis of selected properties of this diagram and of its dual, and generalize the α-complex to this setting. Finally, we provide in Section 3.4 a naive algorithm for computing the dual of the CW Voronoi diagram.

3.2 Toleranced Models and Compoundly Weighted Voronoi Diagram

3.2.1 Compoundly Weighted Distance and Toleranced Balls

Toleranced Balls

Given a collection of weighted points $W_i(c_i; a_i, b_i)$, with center c_i and parameters (real numbers) $a_i > 0$ and b_i, we define the additively-multiplicatively distance between W_i and a point x as follows:

$$\lambda(W_i, x) = a_i \mid\mid c_i - x \mid\mid - b_i. \tag{3.1}$$

This distance is associated with so-called compoundly-weighted Voronoi diagrams [OBSC00]. Geometrically speaking, this distance is best understood using the following growth process. Let a *toleranced ball* $\overline{B_i}(c_i; r_i^-, r_i^+)$ be a pair of concentric balls of radii $r_i^- < r_i^+$, centered at c_i. These balls are called the *inner* and *outer* balls. Given a toleranced ball $\overline{B_i}$ and a real parameter λ, consider the *grown ball* $\overline{B_i}[\lambda]$ centered at c_i and whose radius is defined by:

$$r_i(\lambda) = r_i^- + \lambda(r_i^+ - r_i^-). \tag{3.2}$$

Denoting $\delta_i = r_i^+ - r_i^-$, a point x is reached by this growth process once $r_i(\lambda) = \mid\mid c_i - x \mid\mid$, that is

$$\lambda(\overline{B_i}, x) = \frac{\mid\mid c_i - x \mid\mid}{\delta_i} - \frac{r_i^-}{\delta_i}. \tag{3.3}$$

In other words, a toleranced ball $\overline{B_i}(c_i; r_i^-, r_i^+)$ is tantamount to a weighted point $W_i(c_i; a_i = 1/\delta_i, b_i = r_i^-/\delta_i)$; and reciprocally, a weighted point $W_i(c_i; a_i, b_i)$ is tantamount to a toleranced ball $\overline{B_i}(c_i; r_i^- = b_i/a_i, r_i^+ = (1 + b_i)/a_i)$.
In the sequel, we shall use both terminologies and exchangeably refer to a weighted point W_i or to a toleranced ball $\overline{B_i}$.

3.2.2 On Concomitant Interpolation Processes

Consider two toleranced balls $\overline{B_i}$ and $\overline{B_j}$. We term the linear interpolation of Eq. (3.2) *concomitant* since at $\lambda = 0$ (resp. $\lambda = 1$) the grown balls $\overline{B_i}[\lambda]$ and $\overline{B_j}[\lambda]$ respectively match their inner (outer) balls. Concomitance is important since, for a collection of toleranced balls, we aim at exploring the region sandwiched between the inner and outer balls coherently. Interestingly, concomitance requires multiplicatively weighted Voronoi diagram — CW or Möbius.

Non Concomitant Interpolations

For the power diagram, the growth process consists of modifying the weight w_i (i.e the squared radius r_i^2) as follows:

$$w_i(\alpha) = r_i^2(\alpha) = ||\ c_i - x\ ||^2 = w_i + \alpha \tag{3.4}$$

Let a toleranced weighted point be a pair of concentric balls of weights $w_i = (r_i^-)^2$ and $(r_i^+)^2$. The value b_i required to interpolate from the inner to the outer ball is $b_i = (r_i^+)^2 - (r_i^-)^2$. The interpolation is not concomitant since for two toleranced weighted points, one generically has $b_i \neq b_j$.

The same observation holds for the growth process associated with an Apollonius diagram, which is not concomitant unless the discrepancy $r_i^+ - r_i^-$ of all toleranced balls is equal to some constant.

Concomitant Interpolations

To see that Möbius diagrams share the concomitance property with CW diagrams, recall that the generalized Möbius distance to a weighted point $W_i(c_i, a_i, b_i)$ is defined by:

$$d(W_i, x) = a_i\ ||\ c_i - x\ ||^2 - b_i. \tag{3.5}$$

Equivalently,

$$||\ c_i - x\ ||^2 = \frac{1}{a_i}(d + b_i). \tag{3.6}$$

To make the connexion between the distance of Eq. (3.5) and a toleranced ball, we use $d = 0$ and $d = 1$, which yields

$$(r_i^-)^2 = \frac{b_i}{a_i}, \text{ and } (r_i^+)^2 = \frac{1 + b_i}{a_i}. \tag{3.7}$$

Equivalently, one has:

$$a_i = \frac{1}{(r_i^+)^2 - (r_i^-)^2} \text{ and } b_i = \frac{(r_i^-)^2}{(r_i^+)^2 - (r_i^-)^2}. \tag{3.8}$$

A comparison of the CW and Möbius growth models, that is $r_i(\lambda) = ||\ c_i - x\ ||$ versus $r_i(d) = \sqrt{||\ c_i - x\ ||^2}$, is provided on Figure 3.1. Compared to the CW linear growth model and as shown by the variation of the derivative of $\partial r_i(d)/\partial d$, a large difference $(r_i^+)^2 - (r_i^-)^2$ biases the Möbius interpolation towards small values.

3.2.3 Toleranced Tangency and Generalization of the Empty Ball Property

For affine (Apollonius) Voronoi diagrams, it is well known that for each point centered on a Voronoi face, there exists a unique ball orthogonal (tangent) to the balls associated with the vertices of the dual simplex, and conflict-free with all the other balls [1]. To derive the analogue in the CW-case, consider a point x and two toleranced balls $\overline{B_i}$ and $\overline{B_j}$ such that $\lambda(\overline{B_i}, x) = \lambda < \lambda(\overline{B_j}, x)$. For the pair $\overline{B_i}$ and x, one gets with Eq. (3.1):

$$||\ c_i - x\ || - \frac{b_i}{a_i} - \frac{\lambda}{a_i} = 0 \Leftrightarrow ||\ c_i - x\ || - r_i^- - \lambda\delta_i = 0. \tag{3.9}$$

[1] Consider e.g. the power case, and pick a point x on the Voronoi face dual of a simplex involving a ball $B_i(c_i, w_i)$. Assume that point x lies on the sphere bounding the ball $B_i(c_i, w_i + \alpha)$. One has $\pi(x, B_i) = ||\ c_i - x\ ||^2 - w_i - \alpha = 0$, or equivalently, the balls B_i and $X(x, \alpha)$ are orthogonal.

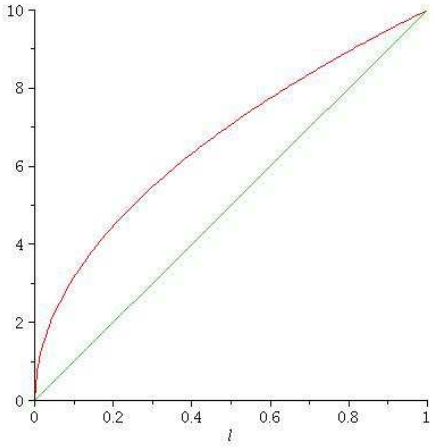


Figure 3.1: Comparing the variation of the radius for the compoundly weighted model (green curve) and the Möbius model (red curve) as a function of the interpolation parameter between 0 and 1. On this example, $r_i^- = 0$ and $r_i^+ = 10$.

Similarly, for the pair $\overline{B_j}$ and point x:

$$|| \ c_j - x \ || - \frac{b_j}{a_j} - \frac{\lambda}{a_j} > 0 \Leftrightarrow || \ c_j - x \ || - r_j^- - \lambda\delta_j > 0. \tag{3.10}$$

We summarize with the following definition, illustrated on Figure3.2:

Definition. 1. *A ball $B(x,\lambda)$ which satisfies the condition of Eq. (3.9) with respect to a toleranced ball $\overline{B_i}$ is called* toleranced tangent *(TT for short) to $\overline{B_i}$. A toleranced ball $\overline{B_j}$ and a ball $B(x,\lambda)$ which satisfy the condition of Eq. (3.10) are called* conflict free.

Remark 1. *Equation (3.9) states that the inner ball $\overline{B_i}[0]$ and the ball $B(x,\lambda\delta_i)$, namely the ball $B(x,\lambda)$* scaled *by δ_i, are tangent. Similarly, condition (3.10) states that $\overline{B_j}[0]$ and $B(x,\lambda\delta_j)$ do not intersect. We shall use this property to illustrate TT balls, see e.g. Figure 3.2.*

Remark 2. *Let $\overline{\mathscr{S}}$ be a collection of toleranced balls. Consider a ball $B(x,\lambda)$ which is TT to a subset of balls $T \subset \overline{\mathscr{S}}$, and conflict-free with the toleranced balls in $\overline{\mathscr{S}} \backslash T$. The center x of this ball is found at the intersection of the spheres bounding the grown balls $\overline{B_i}[\lambda]$ with $\overline{B_i} \in T$, and is located outside the grown balls $\overline{B_j}[\lambda]$ with $\overline{B_j} \in \overline{\mathscr{S}} \backslash T$.*

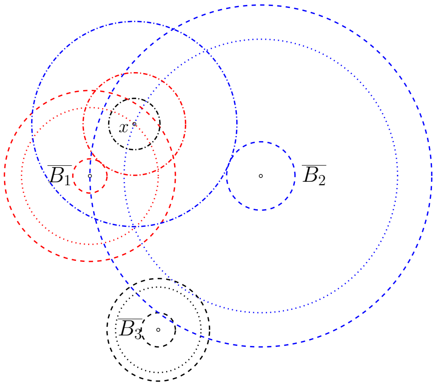


Figure 3.2: Toleranced tangent (TT) balls and conflict-free balls. In dashed lines, toleranced balls $\overline{B_1}(0,0;1,5)$, $\overline{B_2}(0,10;2,8)$, $\overline{B_3}(4,-9;1,3)$. The three dotted circles represent $\overline{B_1}[3/4]$, $\overline{B_2}[3/4]$, $\overline{B_3}[3/4]$. The three circles centered at x are the δ_i-scaled versions of ball $B(x,3/4)$; following remark 1, ball $B(x,3/4)$ is TT to $\overline{B_1}$ and $\overline{B_2}$, and conflict-free with $\overline{B_3}$.

Remark 3. *In Eq. (3.9) and (3.10), the radius of the toleranced ball $B(x, \lambda\delta_i)$ depends on the parameter δ_i from toleranced ball $\overline{B_i}$. Denoting δ^+ the additively weighted distance between two weighted points, Eq. (3.9) and Eq. (3.10) may be rewritten as follows:*

$$\delta^+(B(x,\lambda), \overline{B_i}[0]) = \frac{\delta_i - 1}{\delta_i} \mid\mid c_i - x \mid\mid, \tag{3.11}$$

and

$$\delta^+(B(x,\lambda), \overline{B_i}[0]) > \frac{\delta_i - 1}{\delta_i} \mid\mid c_i - x \mid\mid. \tag{3.12}$$

The left-hand term involves $B(x,\lambda)$, a ball whose radius does not depend on parameters from toleranced balls of $\overline{\mathscr{S}}$, as for the power and Apollonius cases. But the right-hand-side depends on δ_i. In the sequel, we use Eq. (3.9) and Eq. (3.10) for a simpler geometric interpretation of toleranced tangency and conflict-ness. A generic ball not belonging to $\overline{\mathscr{S}}$ will be denoted $B(x,\lambda)$.

3.3 Compoundly Weighted Voronoi Diagrams and Space Filling Diagrams

Consider a collection $\overline{\mathscr{S}}$ of n toleranced balls, and denote $\mathscr{F}_\lambda$ the space-filling diagram, i.e. the domain covered by the grown balls for a given value of λ. The Compoundly Weighted Voronoi diagram is the partition of the space according to the *nearest neighbor* relationship, for the CW distance, that is:

$$Vor(\overline{B_i}) = \{x \in \mathbb{R}^3 \mid \lambda(\overline{B_i}, x) \leq \lambda(\overline{B_j}, x) \forall j \neq i\}. \tag{3.13}$$

More generally, denoting T_{k+1} a tuple of $k+1$ toleranced balls, we are interested in $Vor(T_{k+1}) = \cap_{\overline{B_i} \in T_{k+1}} Vor(\overline{B_i})$. Naturally, we are also interested in the dual complex generalizing the Delaunay triangulation, and in the subset of the dual complex accounting for topological changes of the space-filling diagram $\mathscr{F}_\lambda$.

3.3.1 Bisectors in the CW Case

The bisector of a tuple of toleranced balls T_{k+1} is the loci of points having the same CW distance with respect to every toleranced ball. We denote this bisector $\zeta(T_{k+1})$, and examine in turn the case for pairs, triples, and quadruples. Our analysis assumes that the δ_i are not equal, as this is the Apollonius case [BWY06].

Bisector of two toleranced balls

Analysis. Let $\overline{B_i}$ and $\overline{B_j}$ be two toleranced balls. The ball $\overline{B_i}$ is *trivial* with respect to $\overline{B_j}$ iff $\zeta(i,j)$ does not exist and $\lambda(\overline{B_j}, c_i) < \lambda(\overline{B_i}, c_i)$. The following property describes the triviality of $\overline{B_i}$ w.r.t $\overline{B_j}$:

Proposition. 1. *$\overline{B_i}$ is* trivial *with respect to $\overline{B_j}$ iff $\delta_i \leq \delta_j$ and the following condition, which states that c_i belongs to the interior of the Voronoi region of $\overline{B_j}$, holds:*

$$\lambda(\overline{B_j}, c_i) < -\frac{r_i^-}{\delta_i}. \tag{3.14}$$

Proof. If the Voronoi region V_i of $\overline{B_i}$ is empty, one has in particular, $c_i \notin V_i$, which is exactly Eq. (3.14). The second implication also trivial holds. For the converse, applying the definition of $\lambda(\overline{B_i}, x)$ to any point x, we get:

$$\lambda(\overline{B_i}, x) = \frac{\mid\mid c_i - x \mid\mid - r_i^-}{\delta_i} > \frac{\mid\mid c_i - x \mid\mid}{\delta_i} + \frac{\mid\mid c_i - c_j \mid\mid - r_j^-}{\delta_j} \tag{3.15}$$

$$\geq \frac{\mid\mid c_i - x \mid\mid + \mid\mid c_i - c_j \mid\mid - r_j^-}{\delta_j} \tag{3.16}$$

$$> \frac{\mid\mid c_j - x \mid\mid - r_j^-}{\delta_j} = \lambda(\overline{B_j}, x). \tag{3.17}$$

The three derivations respectively stem from Eq. (3.14), from $\delta_i \leq \delta_j$, and from the triangle inequality. □

Assuming that $\zeta(i,j)$ exists, its geometry depends on the relative values of δ_i and δ_j. Assuming w.l.o.g. that $\delta_i < \delta_j$, $\overline{B_j}$ grows faster than $\overline{B_i}$; for a large enough value of λ, the grown ball $\overline{B_i}[\lambda]$ is contained in its counterpart $\overline{B_j}[\lambda]$, so that the bisector is a closed surface, with c_i in the bounded region delimited by $\zeta(i,j)$. Matching the generalized distances shows that this surface is a degree-four algebraic surface. See Figure 3.3 for a 2D illustration.

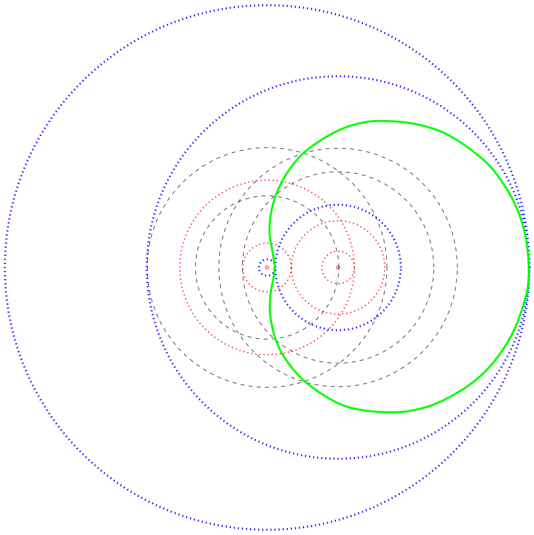


Figure 3.3: Two toleranced balls and their bisector which is a degree four algebraic curve –green curve. Dashed circles corresponding to the inner and outer balls. Dotted circles correspond to the solutions of a degree four equation : blue ones are toleranced tangent circles, red ones are algebraic artifacts.

Extremal TT balls. If the bisector exists, it makes sense to track the TT balls such that the corresponding λ value is a local extremum. By radial symmetry with respect to the line joining the centers of the balls, such balls are necessarily centered at the intersection between the bisector and the line joining the centers. Assume w.l.o.g. that $\delta_i < \delta_j$. The minimal ball satisfying the above condition, denoted $\underline{M}_{i,j}(\underline{m}_{i,j}, \underline{\rho}_{i,j})$ is such that $\overline{B_i}[\underline{\rho}_{i,j}]$ and $\overline{B_j}[\underline{\rho}_{i,j}]$ are tangent at $\underline{m}_{i,j}$. The maximal ball $\overline{M}_{i,j}(\overline{m}_{i,j}, \overline{\rho}_{i,j})$ is such that $\overline{B_i}[\overline{\rho}_{i,j}]$ is interior-tangent to $\overline{B_j}[\overline{\rho}_{i,j}]$ at $\overline{m}_{i,j}$.

Remark 4. *As illustrated on Figure 3.4, $\overline{B_j}[\underline{\rho}_{i,j}]$ may be exterior or interior to $\overline{B_i}[\underline{\rho}_{i,j}]$; the ball $\overline{B_j}[\underline{\rho}_{i,j}]$ is interior to $\overline{B_i}[\underline{\rho}_{i,j}]$ iff $\overline{B_i}$ is closer to c_j than $\overline{B_j}$ for the CW distance i.e. $\lambda(\overline{B_i}, c_j) < \lambda(\overline{B_j}, c_j)$. In the limit case $\lambda(\overline{B_i}, c_j) = \lambda(\overline{B_j}, c_j)$, $c_j = \underline{m}_{i,j}$ and $\overline{B_j}[\underline{\rho}_{i,j}]$ may be considered as exterior to $\overline{B_i}[\underline{\rho}_{i,j}]$.*

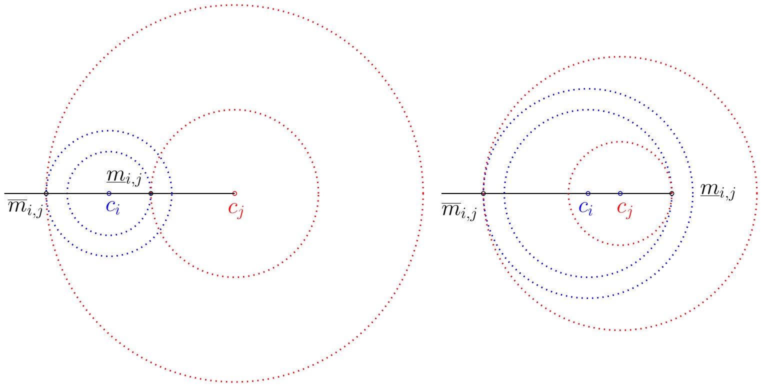


Figure 3.4: Relative position of minimal and maximal TT balls of two balls (Left) $\overline{B_j}[\underline{\rho}_{i,j}]$ and $\overline{B_i}[\underline{\rho}_{i,j}]$ are exterior tangent (Right) $\overline{B_j}[\underline{\rho}_{i,j}]$ is interior tangent to $\overline{B_i}[\underline{\rho}_{i,j}]$.

The parameters of these extremal TT balls are computed as follows:

Proposition. 2. *The two extremal TT balls $B(x,\lambda)$ of two toleranced balls are characterized by*

$$\lambda = \frac{\| \; c_i - c_j \; \| - (\alpha r_i^- + \beta r_j^-)}{\alpha \delta_i + \beta \delta_j}, \tag{3.18}$$

and

$$\overrightarrow{c_i x} = \alpha \frac{\lambda \delta_i + r_i^-}{\| \; c_i - c_j \; \|} \overrightarrow{c_i c_j}, \tag{3.19}$$

where $\alpha = \pm 1$ and $\beta = \pm 1$ depend on the ball processed (minimal or maximal) and the relative positions of $\overline{B_i}$ and $\overline{B_j}$ (case analysis in the proof).

Proof. Denote $\overrightarrow{u}_{x,x'}$ the unit vector between two points x and x'. The extremal TT ball $\underline{M}_{i,j} = (x,\lambda)$ or $\overline{M}_{i,j} = (x,\lambda)$ of $\overline{B_i}$ and $\overline{B_j}$ being centered on the line joining the centers c_i and c_j, we can express the weight λ as follows:

$$\overrightarrow{c_i x} + \overrightarrow{x c_j} = \overrightarrow{c_i c_j} \tag{3.20}$$

$$\Leftrightarrow \| \; c_i - x \; \| \overrightarrow{u}_{c_i x} + \| \; c_j - x \; \| \overrightarrow{u}_{x c_j} = \| \; c_i - c_j \; \| \overrightarrow{u}_{c_i c_j} \tag{3.21}$$

$$\Leftrightarrow \| \; c_i - x \; \| \overrightarrow{u}_{c_i x} . \overrightarrow{u}_{c_i c_j} + \| \; c_j - x \; \| \overrightarrow{u}_{x c_j} . \overrightarrow{u}_{c_i c_j} = \| \; c_i - c_j \; \| \tag{3.22}$$

$$\alpha(\lambda \delta_i + r_i^-) + \beta(\lambda \delta_j + r_j^-) = \| \; c_i - c_j \; \|, \tag{3.23}$$

where $\alpha = \overrightarrow{u}_{c_i x} . \overrightarrow{u}_{c_i c_j} = \pm 1$ and $\beta = \overrightarrow{u}_{x c_j} . \overrightarrow{u}_{c_i c_j} = \pm 1$. Equation (3.18) follows easily. We note in passing that following remark 4, the signs of the dot products α and β are obtained from the sign of the expression $\lambda(\overline{B_i}, c_j) - \lambda(\overline{B_j}, c_j)$.

The weight of the extremal TT balls being determined, the center is computed as follows:

$$\alpha \overrightarrow{u}_{c_i x} = \overrightarrow{u}_{c_i c_j} \tag{3.24}$$

$$\Leftrightarrow \alpha \frac{\overrightarrow{c_i x}}{\| \; c_i - x \; \|} = \frac{\overrightarrow{c_i c_j}}{\| \; c_i - c_j \; \|} \tag{3.25}$$

$$\Leftrightarrow \overrightarrow{c_i x} = \alpha \frac{\| \; c_i - x \; \|}{\| \; c_i - c_j \; \|} \overrightarrow{c_i c_j} \tag{3.26}$$

$$\Leftrightarrow \overrightarrow{c_i x} = \alpha \frac{\lambda \delta_i + r_i^-}{\| \; c_i - c_j \; \|} \overrightarrow{c_i c_j}. \tag{3.27}$$

□

Bisector of three toleranced balls

Analysis. Consider three toleranced balls $\overline{B_{i_0}}, \overline{B_{i_1}}, \overline{B_{i_2}}$ such that the bisector of each pair exists. To avoid the Apollonius case, we suppose without loss of generality that $\delta_{i_0} \leq \delta_{i_1} \leq \delta_{i_2}$ with $\delta_{i_0} < \delta_{i_2}$. If there is no intersection between $\zeta(i_0,i_1)$ and $\zeta(i_0,i_2)$, $\zeta(i_0,i_1,i_2)$ does not exist, and reciprocally. Assume that $\zeta(i_0,i_1,i_2)$ exists. Since at least one δ_i differs from the other two, there is at most one Apollonius bisector. The geometry of $\zeta(i_0,i_1,i_2)$ depends on $\delta_{i_0}, \delta_{i_1}$ and δ_{i_2}, and the following cases are illustrated on Figure 3.5.

▷ **CWB.III.1** If there is no Apollonius bisector, $\zeta(i_0,i_1,i_2)$ is a bounded curve resulting from the intersection of two CW bisectors.

▷ **CWB.III.2** If the Apollonius bisector is not a half straight line, $\zeta(i_0,i_1,i_2)$ is a bounded curve resulting from the intersection of one CW bisector, and one sheet of a hyperboloid (possibly degenerated to a hyperplane).

▷ **CWB.III.3** If the Apollonius bisector is a half straight line, $\zeta(i_0,i_1,i_2)$ is reduced to at most two intersection points. Note that if there are two intersection points, $\delta_{i_1} = \delta_{i_2}$ and $\overline{B_{i_1}}$ is included in and tangent to $\overline{B_{i_2}}$.

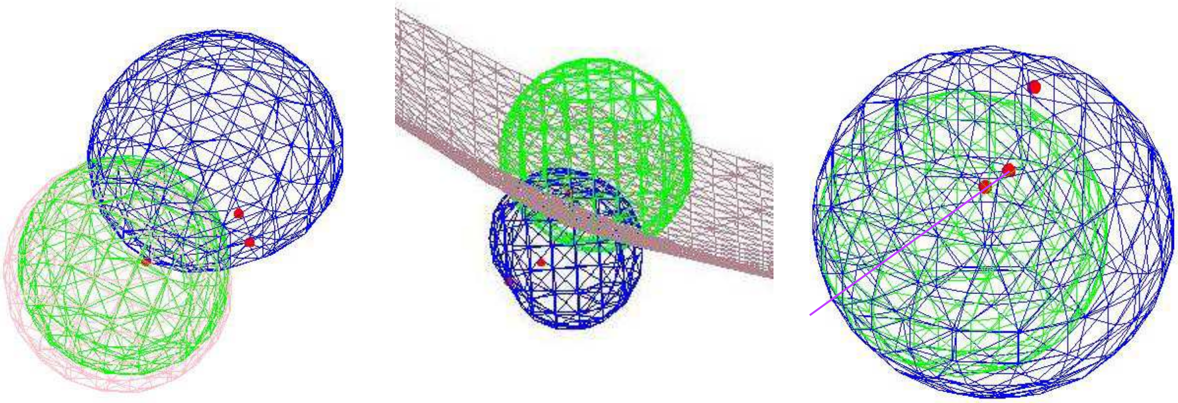


Figure 3.5: Bisectors of three toleranced balls. The red dots are the centers of the toleranced balls and the pink/green/blue surfaces respectively represent the bisectors $\zeta(0,1)$ / $\zeta(0,2)$ / $\zeta(1,2)$. (Left) **CWB.III.1** No Apollonius bisector (Middle) **CWB.III.2** One Apollonius bisector (Right) **CWB.III.3** One degenerate Apollonius bisector.

Extremal TT balls. In any case, there are two (possibly identical) extremal TT balls. If one bisector is a half straight line, these balls are found by intersecting this line with one of the other two bisectors.
In the general case, identifying these two balls involves four equations in four unknowns—the coordinates of the center and the weight λ. Denote π the plane defined by the centers of the three balls. The growth of the balls being symmetric with respect to this plane, the fourth equation consists of constraining the center of an extremal TT ball to plane π. The calculation is covered by the following proposition for $k = 2$:

Proposition. 3. *Let $T_{k+1} = \{\overline{B_{i_j}}\}_{j=0,\ldots,k}$ be a triple or quadruple of toleranced balls, i.e. $k = 2$ or $k = 3$. Computing the two extremal TT balls of the tuple T_{k+1} requires solving a degree four equation. A solution value λ of this equation is valid provided that $\lambda\delta_{i_j} + r^-_{i_j} \geq 0$, $\forall j = 0,\ldots,k$.*

Proof. of Prop. 3 for $T_{k+1} = \{\overline{B_{i_0}}, \overline{B_{i_1}}, \overline{B_{i_2}}\}$.
The ball sought has to be TT to each of the three toleranced balls, as specified by Eq. (3.9). Let P be the plane containing the centers of the three balls. Squaring the three equations of toleranced tangency yields the system:

$$\begin{cases} (x - c_{i_0})^2 = (\lambda\delta_{i_0} + r^-_{i_0})^2 \\ (x - c_{i_1})^2 = (\lambda\delta_{i_1} + r^-_{i_1})^2 \\ (x - c_{i_2})^2 = (\lambda\delta_{i_2} + r^-_{i_2})^2 \\ P[x] = 0 \end{cases} \tag{3.28}$$

Subtracting the first squared equation from the two subsequent ones yields:

$$\begin{cases} (x-c_{i_0})^2 = (\lambda\delta_{i_0} + r^-_{i_0})^2 \\ 2x(c_{i_1} - c_{i_0}) = (\lambda\delta_{i_0} + r^-_{i_0})^2 - (\lambda\delta_{i_1} + r^-_{i_1})^2 - (c^2_{i_0} - c^2_{i_1}) \\ 2x(c_{i_2} - c_{i_0}) = (\lambda\delta_{i_0} + r^-_{i_0})^2 - (\lambda\delta_{i_2} + r^-_{i_2})^2 - (c^2_{i_0} - c^2_{i_2}) \\ P[x] = 0 \end{cases} \tag{3.29}$$

Using Gaussian elimination on the last three equations, one obtains three linear equations for the coordinates of x, parametrized by λ^2. Injecting these quantities into the first equation yields the quartic equation in λ. Note that a solution is valid iff $\lambda\delta_{i_j} + r^-_{i_j} \geq 0$, and sorting the valid values yields the extreme TT balls. Also note that the coordinates of point x are rational fractions in λ^2.

□

Remark 5. *Geometrically, three intersecting spheres generically intersect in two points. The extreme TT balls correspond to the situations where these two points coalesce.*

Bisector of four toleranced balls

Analysis. Consider four toleranced balls $\overline{B_{i_0}}, \overline{B_{i_1}}, \overline{B_{i_2}}, \overline{B_{i_3}}$ such that the bisector of each pair exists. To avoid the Apollonius case, we suppose w.l.o.g. that $\delta_{i_0} \leq \delta_{i_1} \leq \delta_{i_2} \leq \delta_{i_3}$ with $\delta_{i_0} < \delta_{i_3}$. If the intersection of $\zeta(i_0, i_1)$, $\zeta(i_0, i_2)$ and $\zeta(i_0, i_3)$ is empty, the intersection of all bisectors of pairs is empty and $\zeta(i_0, i_1, i_2, i_3)$ does not exist, and reciprocally. If $\zeta(i_0, i_1, i_2, i_3)$ exists, we have $\zeta(i_0, i_1, i_2, i_3) = \zeta(i_0, i_3) \cap \zeta(i_1, i_2, i_3)$, from which the following analysis follows.

▷ **CWB.IV.1** The bisectors $\zeta(i_0, i_3)$ and $\zeta(i_1, i_2, i_3)$ being a surface and a curve, their generic intersection, if any, consists of a finite set of points. As we shall see below, there are at most four such points.

▷ **CWB.IV.2** As a degenerate case, when $\zeta(i_1, i_2, i_3)$ is a bounded curve, the intersection of $\zeta(i_1, i_2, i_3)$ and $\zeta(i_0, i_3)$ may be $\zeta(i_1, i_2, i_3)$. In this case, $\zeta(i_0, i_1, i_2, i_3)$ has the geometry of the bisector $\zeta(i_1, i_2, i_3)$ of three toleranced balls.

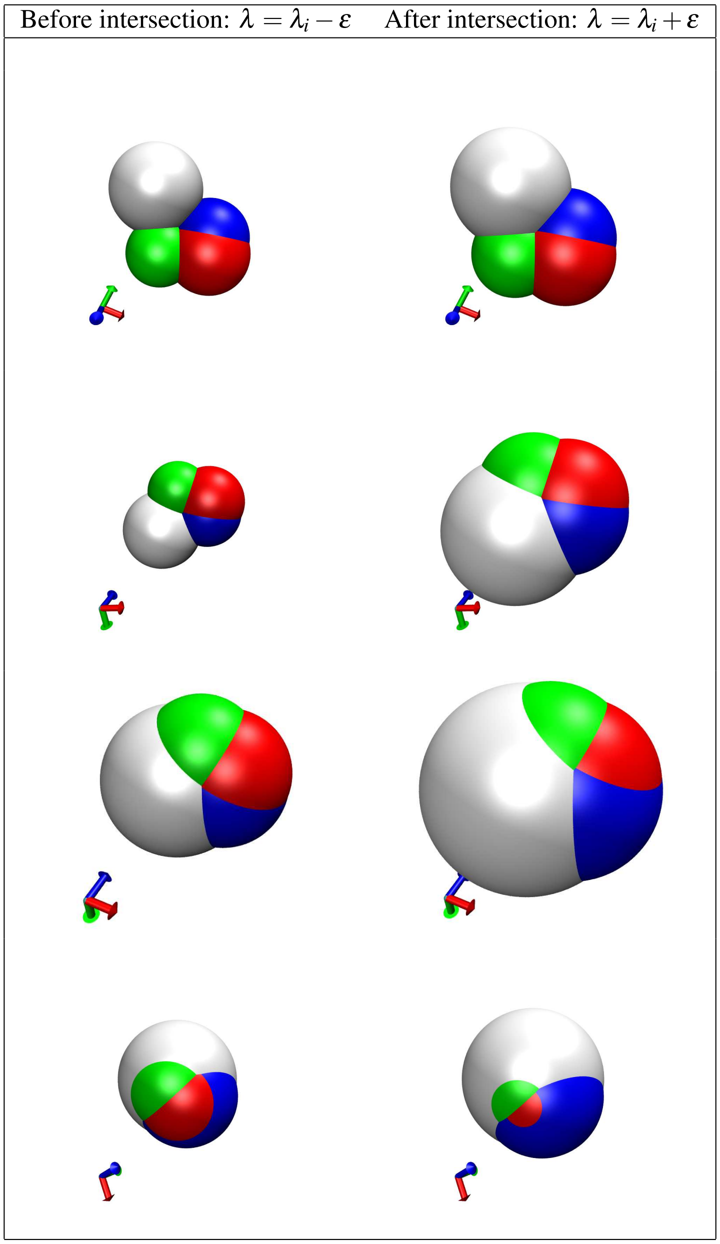
Extremal TT balls. We distinguish two cases. If $\zeta(i_0, i_1, i_2, i_3)$ has the geometry of a bisector of three toleranced balls, we refer to the analysis carried out in section 3.3.1. Otherwise, $\zeta(i_0, i_1, i_2, i_3)$ is reduced to at most four points, as shown by the following constructive proof of proposition 3:

Proof. of Prop. 3 for $T_{k+1} = \{\overline{B_{i_0}}, \overline{B_{i_1}}, \overline{B_{i_2}}, \overline{B_{i_3}}\}$.
The ball sought has to be TT to each of the four toleranced balls, that is $\|\ c_{i_j} - x\ \| = \lambda\delta_{i_j} + r^-_{i_j}$ for $j = 0, 1, 2, 3$. Squaring the four equations of toleranced tangency yields the system

$$\begin{cases} (x-c_{i_0})^2 = (\lambda\delta_{i_0} + r^-_{i_0})^2 \\ (x-c_{i_1})^2 = (\lambda\delta_{i_1} + r^-_{i_1})^2 \\ (x-c_{i_2})^2 = (\lambda\delta_{i_2} + r^-_{i_2})^2 \\ (x-c_{i_3})^2 = (\lambda\delta_{i_3} + r^-_{i_3})^2 \end{cases} \tag{3.30}$$

As for the case of three toleranced balls, we use Gaussian elimination on system (3.30) to get three equations linear to the coordinates of x and parametrized by λ^2, and one quartic equation on λ. Checking that $\|\ c_{i_j} - x\ \| \geq 0$ provides the valid solutions, and sorting these provides the extremal solutions. □

Remark 6. *As illustrated on Figure 3.6, the system (3.30) may have four distinct solutions λ_i such that $\lambda_i\delta_j + r^-_i \geq 0$.*



Before intersection: $\lambda = \lambda_i - \varepsilon$ After intersection: $\lambda = \lambda_i + \varepsilon$

Figure 3.6: Upon growing, four toleranced balls may intersect into four distinct points. Denoting ε an arbitrarily small number, we display the toleranced balls $\overline{B_i}[\lambda_j \pm \varepsilon]$. The λ_j have been sorted by increasing value from **Top** to **Bottom**.

3.3.2 Compoundly Weighted Voronoi Diagram and its Dual Complex

Empty Voronoi Regions

A toleranced ball whose region is empty is called *trivial.* Proposition 1 gives a condition of triviality for two toleranced balls. But triviality of a toleranced ball amidst a collection of balls is more complex since a toleranced ball might not be trivial with respect to any other one, yet, it might be trivial with respect to their union. To see why, observe that Eq. (3.13) tells us that a point in space is attributed to the Voronoi region of a toleranced ball provided that this toleranced ball *reaches* this point first in the growth process. Thus, a growing ball which is always contained in the union of a collection of growing balls is trivial, although it might not be trivial with any of them. Denoting T a collection of toleranced balls, and for any value of λ, the following condition, illustrated on

Figure 3.8, must hold for $\overline{B_i}$ to be trivial:

$$\overline{B_i}[\lambda] \subset \bigcup_{\overline{B_j}\in T} \overline{B_j}[\lambda]. \tag{3.31}$$

Remark 7. *The triviality condition is more complex than in the Apollonius case, where a ball is hidden if and only if it is included within another ball.*

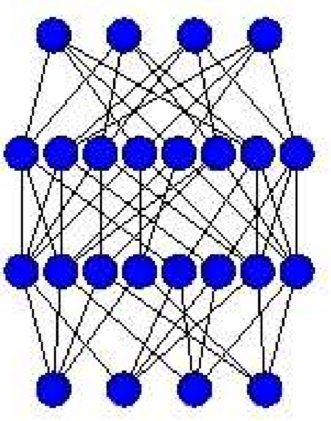


Figure 3.7: Dual complex of the four balls of Figure 3.6—bottom and top rows respectively represent 0-simplices and 3-simplices.

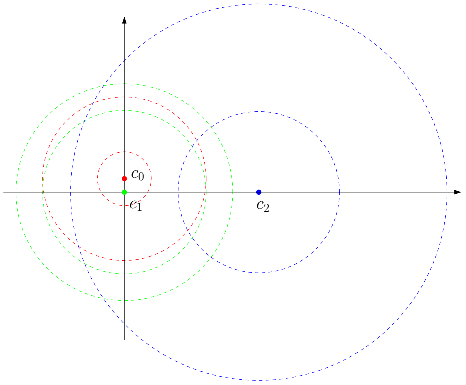


Figure 3.8: Hidden toleranced ball. $\overline{B_0} = (0,1/2;1,3)$ (red), $\overline{B_1} = (0,0;3,4)$ (green) and $\overline{B_2} = (5,0;3,7)$ (blue). Ball $\overline{B_0}$ is neither trivial with respect to $\overline{B_1}$ nor $\overline{B_2}$, but is trivial with respect to both.

Dual Complex

The Voronoi region $Vor(T_{k+1})$ of a tuple T_{k+1} may have several connected components, each being termed a *face*. Each such face corresponds to the intersection of $k+1$ Voronoi regions, so that we associate an *abstract simplex* or *simplex* for short in the dual complex. That is, if $Vor(T_{k+1})$ consists of m faces, one finds $\Delta_j(T_{k+1}), j \in 1,\ldots,m$ simplices in the dual complex. (The multiplicity is omitted if the tuple T_{k+1} yields a single simplex.) The dual of a simplex $\Delta(T)$ is denoted $\Delta(T)^*$. Assuming that the input toleranced balls are numbered from 1 to n, a simplex is *identified* by a list of integers, and inclusion between such lists defines a partial order on simplices. We therefore represent the dual complex by a Hasse diagram D_S with one node per simplex. The nodes of D_S corresponding to k-simplices are denoted $D_S(k)$. Note that we may also (arbitrarily) embed a simplex within the union of Voronoi faces associated to it. See Figs. 3.9, 3.10 and 3.11 for a 2D illustration.

Topological Complications

A Voronoi region gets sandwiched between two neighbors when the corresponding toleranced ball defines a *lens* between the Voronoi region of two neighboring toleranced balls, a case also found in the Apollonius diagram. In the dual complex, the vertex of this toleranced balls has exactly two neighbors and the triangle corresponding to these three toleranced balls does not have any coface.

A Voronoi region might not be connected, and this may happen for tuples of size one to four. We illustrate this in 2D with Figure 3.9. For a toleranced ball, consider $\overline{B_4}$ whose Voronoi region is split into two faces, associated with the vertices (zero-dimensional simplices) $\Delta_1(4)$ and $\Delta_2(4)$ in the Hasse diagram. For two toleranced balls, note that the Voronoi region $Vor(\overline{B_1},\overline{B_2})$ consists of two faces—open line segments in this case, yielding the simplices $\Delta_1(1,2)$ and $\Delta_2(1,2)$ in the Hasse diagram. For three toleranced balls, note that the triple $(\overline{B_1},\overline{B_2},\overline{B_4})$ corresponds to two triangles.

A Voronoi region may not be simply connected. When one toleranced ball punches a hole into a face, the corresponding one-simplex does not have any coface. See e.g. toleranced ball $\overline{B_7}$ and the simplex $\Delta(2,7)$ on Figure 3.9. When two toleranced balls punch a hole into a Voronoi region, the two-simplex they define does not have any coface either. Finally, when three toleranced balls punch a hole into a Voronoi region, two tetrahedra of the dual complex share the same vertices, the same edges and same triangles. This latter case is illustrated in 2D,

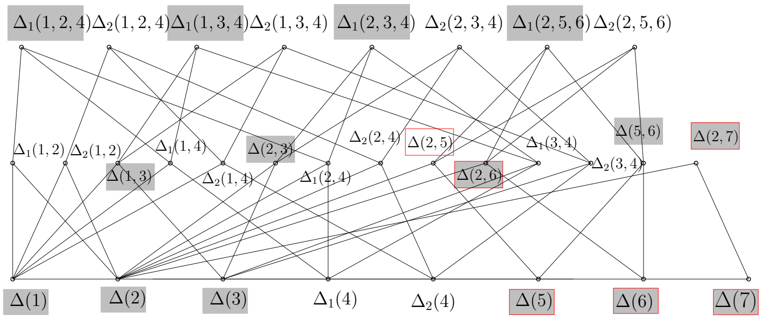


Figure 3.11: Hasse diagram of simplices of the dual complex of the Voronoi diagram of Figure 3.9. The three lines respectively corresponding to 0-simplices, 1-simplices, and 2-simplices. Grey boxes correspond to Gabriel simplices, and boxes with a red boundary mark dominated simplices. See text for details.

where a hole punched by two toleranced balls in a Voronoi region results in two triangles with the same vertices and the same edges. See $\Delta_1(2,5,6)$ and $\Delta_2(2,5,6)$ on Figure 3.9.

Bounded and Unbounded Voronoi Regions

A toleranced ball $\overline{B_i} \in \overline{\mathscr{S}}$ is called *maximal* with respect to $\overline{\mathscr{S}}$ if $\delta_i \geq \delta_j, \forall j \neq i$. A toleranced ball which is not maximal has a bounded Voronoi region in the CW VD of $\overline{\mathscr{S}}$, and the subset of maximal toleranced balls is denoted $\overline{\mathscr{S}}_{\text{max}}$. The CW VD diagram of toleranced balls in $\overline{\mathscr{S}}_{\text{max}}$ is an Apollonius diagram since all δ_i are equal, and a subset of balls in $\overline{\mathscr{S}}_{\text{max}}$ have an unbounded Voronoi region. Mimicking the affine case, a simplex is said to lie on the convex hull $CH(\overline{\mathscr{S}})$ of the dual complex if its dual Voronoi face is unbounded. The vertices of such simplices belong to $\overline{\mathscr{S}}_{\text{max}}$.

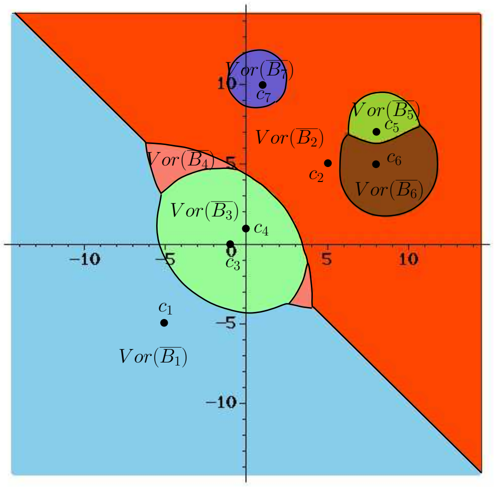


Figure 3.9: The CW VD of 7 toleranced balls in 2D: $\overline{B_1} = (-5,-5;3,7)$, $\overline{B_2} = (5,5;3,7)$, $\overline{B_3} = (-1,0;4,5)$, $\overline{B_4} = (0,1;2,5)$, $\overline{B_5} = (8,7;2,3)$, $\overline{B_6} = (8,5;3,4)$, $\overline{B_7} = (1,10;1,2)$. $Vor(\overline{B_4})$ and $Vor(\overline{B_1},\overline{B_2})$ are not connected. $Vor(\overline{B_2})$ is not simply connected. δ_1 and δ_2 are maximal and $\overline{B_1},\overline{B_2}$ have unbounded Voronoi regions.

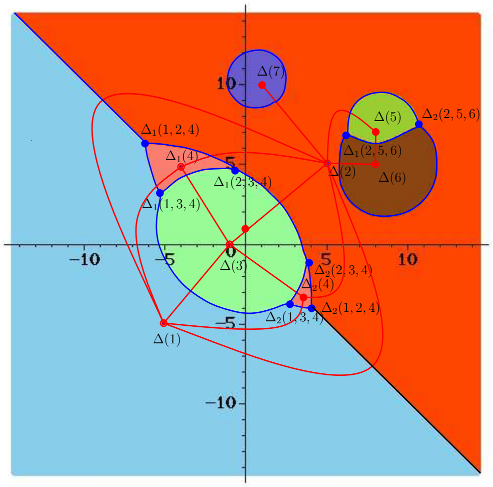


Figure 3.10: Dual complex for the CW VD of Figure 3.9 : 0-simplices: black dots; one-simplices: blue curves; two simplices: red dots. Note that $\overline{B_4}$ is represented by two vertices $\Delta_1(4)$ and $\Delta_2(4)$. $\Delta_1(2,5,6)$ and $\Delta_2(2,5,6)$ share the three same edges. $\Delta(2,7)$ does not bound any triangle.

3.3.3 Gabriel, Dominant and Dominated simplices

In the affine case, changes in the α-complex are associated with Gabriel simplices: a Gabriel simplex $\Delta(T)$ is a simplex such that its minimal orthogonal ball $\underline{M}_T$ is conflict-free, and the simplex enters the α-complex when $\lambda \geq \underline{\rho}_T$, with $\underline{\rho}_T$ the weight of $\underline{M}_T$. The generalization to the CW setting is not straightforward since Voronoi regions might not be connected, and since a tuple T generally has two extremal TT balls, respectively denoted $\underline{M}_T(\underline{m}_T, \underline{\rho}_T)$ and $\overline{M}_T(\overline{m}_T, \overline{\rho}_T)$. We now examine these two balls and refer the reader to Figure 3.12 for an illustration. (To examine this figure, recall that a TT ball $M(x,\lambda)$ is conflict-free with a toleranced ball $\overline{B_i}$ iff the scaled version of M by δ_i i.e. $M(x,\delta_i\lambda)$ does not intersect the inner ball of $\overline{B_i}$.)

Minimal TT Balls and Gabriel Simplices

If the center $\underline{m}_T$ of the minimal TT ball $\underline{M}_T$ belongs to the relative interior of a Voronoi face of the tuple, or equivalently the ball is conflict free, the simplex is called *Gabriel*.

Remark 8. *The minimal TT ball $\underline{M}_T$ is unique, so that a single Voronoi face dual of the tuple T can witness a Gabriel simplex. In particular, for any other Voronoi face of the tuple, the minimal TT ball associated with that face involves at least another toleranced ball. For example, the Voronoi region of $\overline{B_4}$ in Figure 3.9 is split into two Voronoi faces. The center of the minimal TT ball of $\overline{B_4}$ being located within the Voronoi region of $\overline{B_3}$, the dual of each Voronoi face of $\overline{B_4}$ is not Gabriel. The minimal TT balls of these two faces are in fact associated to the same triple, namely $\overline{B_2}$, $\overline{B_3}$ and $\overline{B_4}$.*

Maximal TT Balls and Domination of Simplices

For a simplex $\Delta(T)$, consider the intersection of the spheres bounding the grown balls, i.e.

$$I_T[\lambda] = \cap_{\overline{B_i}\in T} \partial \overline{B_i}[\lambda]. \tag{3.32}$$

In 3D, if T is a $k+1$ tuple, $I_T[\lambda]$ is generically a $2-k$-sphere. Consider now a tuple such that the δ_i of its balls are not all equal, and assume that $T = A \cup B$, with A the balls realizing the maximum δ_i in the tuple. The corresponding bisector is bounded and has a unique maximal TT ball $\overline{M}_T$. If this ball is conflict-free, the spheres bounding the grown balls in T intersect until $I_T[\lambda]$ reduces to the point $\overline{m}_T$. Beyond that point, the intersection of the spheres bounding grown balls in B is contained in the union of the grown balls in A. To formalize this behavior, we define—recall that an ancestor of a node in the Hasse diagram is any node found on a path joining this node to that associated with a zero-dimensional simplex:

Definition. 2. *A simplex $\Delta(T)$ whose dual Voronoi face contains the center $\overline{m}_T$ of the maximal TT ball is called* dominant.
A simplex $\Delta(U)$ which is an ancestor of the dominant simplex $\Delta(T)$ in the Hasse diagram, with $B \subset U \subsetneq T$, is called dominated.

As opposed to the Euclidean setting, a dominant simplex $\Delta(T)$ does not catch a coface when point $\overline{m}_T$ is reached by the growing balls. Similarly, a dominated simplex does not catch any coface either when point $\overline{m}_T$ is reached. To identify the moment in time where simplex $\Delta(U)$ gets dominated, we introduce

$$\gamma_{\Delta(U)} = \overline{\rho}_{\Delta(T)}. \tag{3.33}$$

The condition $B \subset U \subsetneq T$ actually yields 2 cases, namely (i) $U = B$, or (ii) $B \subsetneq U \subsetneq T$. In three dimensions, enumerating these possibilities yields the following cases:

▷ **Dom.1** $T = \{\overline{B_1}, \overline{B_2}\}$ with $\delta_1 > \delta_2$: case (i) that is $U = \{\overline{B_2}\}$.
▷ **Dom.2** $T = \{\overline{B_1}, \overline{B_2}, \overline{B_3}\}$ with $\delta_1 > \delta_2 \geq \delta_3$: case (i) that is $U = \{\overline{B_2}, \overline{B_3}\}$.
▷ **Dom.3** $T = \{\overline{B_1}, \overline{B_2}, \overline{B_3}\}$ with $\delta_1 = \delta_2 > \delta_3$: case (i) that is $U = \{\overline{B_3}\}$, and case (ii) that is $U = \{\overline{B_1}, \overline{B_3}\}$ or $U = \{\overline{B_2}, \overline{B_3}\}$.

By convention and since a 4-tuple yields a discrete set of at most four tetrahedra, we say that a 3-simplex cannot be dominant—which prevents a 2-simplex from being dominated. Also, a 0-simplex cannot be dominant. Dominant and dominated simplices are important to describe the evolution of the boundary $\partial\mathscr{F}_\lambda$: upon getting dominated, a simplex does not contribute to $\partial\mathscr{F}_\lambda$ anymore.

Remark 9. *A dominant simplex may have cofaces.*

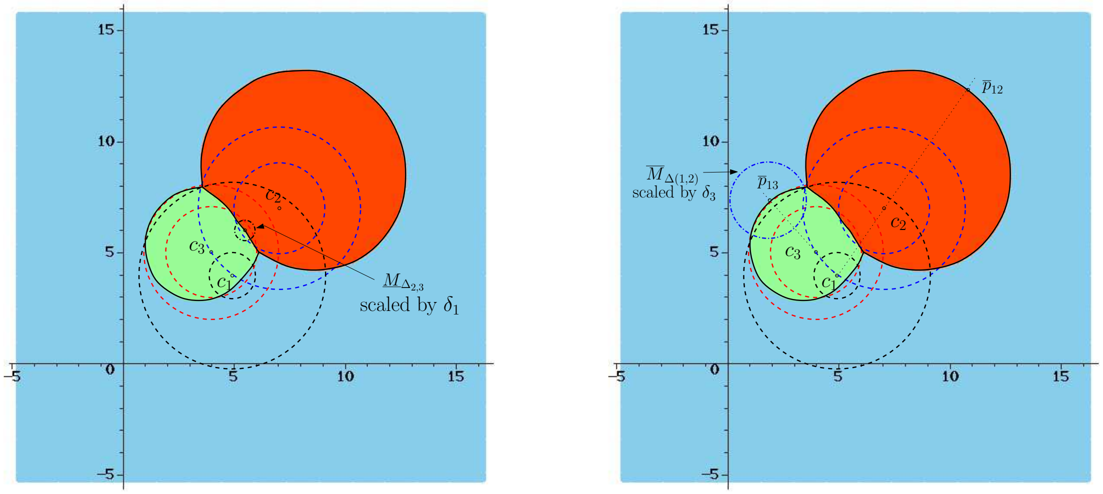


Figure 3.12: Gabriel, dominant and dominated simplices illustrated with the CW VD of 3 toleranced balls (dashed lines): $\overline{B_1} = (5,4;1,4)$ (black), $\overline{B_2} = (7,7;2,3.5)$ (blue), $\overline{B_3} = (4,5;2,3)$ (red). (Left) The minimal TT ball $\underline{M}_{\Delta(2,3)}$ is conflict-free (witnessed by the black dashed-dotted circle): $\Delta(2,3)$ is Gabriel. Simplices $\Delta(2)$, $\Delta(3)$ and $\Delta(1,3)$ are Gabriel too. (Right) The max TT ball $\overline{M}_{\Delta(1,3)}$ is conflict-free (witnessed by the blue dashed-dotted circle): $\Delta(1,3)$ is dominant and $\Delta(3)$ is dominated. $\Delta(1,2)$ is dominant and $\Delta(2)$ is dominated too.

3.3.4 The λ-complex Filtration

The Filtration

Equipped with Gabriel simplices, the following mimics the Euclidean setting:

Definition. 3. *The λ-complex K_λ is a subset of the dual complex defined as follows: a simplex $\Delta(T)$ belongs to K_λ iff (i) $\Delta(T)$ is Gabriel and $\lambda \geq \underline{\rho}_{\Delta(T)}$, or (ii) $\Delta(T)$ is a face of $\Delta(U)$ with $\Delta(U) \in K_\lambda$.*

Increasing λ results in a nested sequence of (abstract) simplicial complexes, which eventually coincide with the dual complex, so that the collection of λ-complexes forms a filtration. At the far left of the spectrum, the first non empty simplicial complex (generically) consists of a dual vertex which appears at $\lambda = \Lambda_{\min}$ defined by:

$$\Lambda_{\min} = \min_{B_i \in \mathscr{S}} \left(-\frac{r_i^{-}}{\delta_i}\right). \tag{3.34}$$

For a large enough λ, the λ-complex matches the dual complex. Since there may be no new Gabriel simplex in the last λ-complex, this holds for $\lambda \geq \Lambda_{\max}$ with

$$\Lambda_{\max} = \max\{\max_{\Delta \text{ Gabriel}} (\underline{\rho}_\Delta), \max_{\Delta \text{ dominant}} (\overline{\rho}_\Delta)\}. \tag{3.35}$$

Remark 10. *Consider Eq. (3.35). If the last event in the λ-complex does not correspond to the addition of a Gabriel simplex, it actually corresponds to a status change, namely a dominant simplex becomes Interior. See Table 3.1.*

Remark 11. *From a computational standpoint, collecting all the simplices of the λ-complex may be an overkill. In particular, contacts between proteins may be studies resorting to (abstract) simplices of dimension one. To the end, we define the* partial *λ-complex as the subset of the λ-complex containing only the Gabriel simplices of* *dimension* 0 *and* 1. *Computing the partial λ-complex is a straightforward task. A naive algorithm indeed consists of visiting all the possible pairs of toleranced balls* $(\overline{B_i}, \overline{B_j})$*, and to check that* $\underline{M}_{i,j}$ *is conflict-free with all other toleranced balls. For n toleranced balls, this is done in* $O(n^3)$.

Status of Simplices

The status of a simplex in the affine setting is described from the topology of its link. In the CW case, the presence of simplices without any coface as described in section 3.3.2 requires devising different classification criteria. For a simplex $\Delta(T)$, consider the intersection $I_T[\lambda]$ of Eq. (3.32). Upon increasing λ, this intersection sweeps the Voronoi region of the tuple. We base our classification on the portion of the Voronoi region swept by $I_T[\lambda]$ up to time λ. That is, a k-simplex of K_λ is classified as follows:
– *Singular*: the region swept by $I_T[\lambda]$ up to time λ is contained in the relative interior of the dual of the simplex.
– *Interior*: the region swept by $I_T[\lambda]$ up to time λ contains the dual of the simplex in its interior.
– *Regular*: neither singular nor interior.

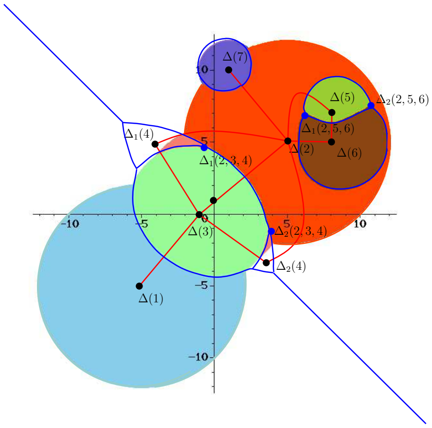


Figure 3.13: Restricted Voronoi regions for the CW VD of Figure 3.9 for $\lambda = 1$, and classification of edges in the λ-complex. Classification of the 8 dual vertices—black dots: $\Delta(5)$ and $\Delta(6)$ are interior and all other dual vertices are regular. Classification of the 10 dual edges—blue edges: $\Delta(1,3)$ and $\Delta(2,7)$ are singular; $\Delta(2,3)$, $\Delta(2,5)$, $\Delta(2,6)$, $\Delta(5,6)$ are interior; the remaining edges are regular. The 4 dual triangles in the λ-complex are represented by red dots vertices.

3.3.5 Classification of Simplices

Our classification of simplices follows the framework of the affine case [Ede92]. For simplices which are neither dominant nor dominated, in addition to the weight $\underline{\rho}_{\Delta(T)}$ of the minimal TT ball, we denote $\underline{\mu}_{\Delta(T)}$ and $\overline{\mu}_{\Delta(T)}$ the λ-values such that the simplex becomes regular and interior. For dominant simplices, we also use the weight of the maximal TT ball $\overline{\rho}_{\Delta(T_)}$, and the quantity $\gamma_{\Delta(U)}$ introduced in Eq. (3.33).

For the affine α-complex, the classification of a simplex as singular, regular, or interior requires considering the four cases { Gabriel, not Gabriel } × { on the convex hull, not on the convex hull }. For simplices which are neither dominant nor dominated, these four possibilities are also found in the CW case—lines 1-4 of Table 3.1. On the other hand, dominant and dominated simplices are not found on the convex hull—each such simplex involves at least one non-maximal ball, and always end up interior since the maximal TT ball of the tuple of a dominant simplex is conflict-free. For dominant simplices, the two additional cases to be considered are Gabriel and non Gabriel—lines 5-6 in Table 3.1. Such a simplex becomes interior as soon as $\lambda \geq \overline{\rho}_{\Delta(T)}$.
Similarly for dominated simplices, the two additional cases to be considered are Gabriel and non Gabriel—lines 7-8 in Table 3.1. Recall that a dominated simplex is always associated to a dominant simplex. Using the weight $\gamma_{\Delta(T)}$ of the maximal TT ball of the tuple of the dominant simplex associated to the dominated simplex, see Eq. (3.33), the dominated simplex becomes interior as soon as $\lambda \geq \gamma_{\Delta(T)}$.

These notions are illustrated on Figure 3.13, which features the restricted Voronoi diagram, i.e. the grown balls restricted to their Voronoi regions. Note in particular that the status of simplices reads from the relative position of the restriction with respect to the associated Voronoi face, as specified in section 3.3.4.

	singular	regular	interior
(1) $\Delta(T) \in CH(\overline{\mathscr{S}})$,Gabriel, non dominated/dominant	$(\underline{\rho}_{\Delta(T)}, \underline{\mu}_{\Delta(T)}]$	$(\underline{\mu}_{\Delta(T)}, +\infty]$	
(2) $\Delta(T) \in CH(\overline{\mathscr{S}})$,non Gabriel, non dominated/dominant		$(\underline{\mu}_{\Delta(T)}, +\infty]$	
(3) $\Delta(T) \notin CH(\overline{\mathscr{S}})$ Gabriel, non dominated/dominant	$(\underline{\rho}_{\Delta(T)}, \underline{\mu}_{\Delta(T)}]$	$(\underline{\mu}_{\Delta(T)}, \overline{\mu}_{\Delta(T)}]$	$(\overline{\mu}_{\Delta(T)}, +\infty]$
(4) $\Delta(T) \notin CH(\overline{\mathscr{S}})$,non Gabriel, non dominated/dominant		$(\underline{\mu}_{\Delta(T)}, \overline{\mu}_{\Delta(T)}]$	$(\overline{\mu}_{\Delta(T)}, +\infty]$
(5) $\Delta(T) \notin CH(\overline{\mathscr{S}})$ Gabriel, dominant	$(\underline{\rho}_{\Delta(T)}, \underline{\mu}_{\Delta(T)}]$	$(\underline{\mu}_{\Delta(T)}, \overline{\rho}_{\Delta(T)}]$	$(\overline{\rho}_{\Delta(T)}, +\infty]$
(6) $\Delta(T) \notin CH(\overline{\mathscr{S}})$,non Gabriel, dominant		$(\underline{\mu}_{\Delta(T)}, \overline{\rho}_{\Delta(T)}]$	$(\overline{\rho}_{\Delta(T)}, +\infty]$
(7) $\Delta(T) \notin CH(\overline{\mathscr{S}})$ Gabriel, dominated	$(\underline{\rho}_{\Delta(T)}, \underline{\mu}_{\Delta(T)}]$	$(\underline{\mu}_{\Delta(T)}, \gamma_{\Delta(T)}]$	$(\gamma_{\Delta(T)}, +\infty]$
(8) $\Delta(T) \notin CH(\overline{\mathscr{S}})$,non Gabriel, dominated		$(\underline{\mu}_{\Delta(T)}, \gamma_{\Delta(T)}]$	$(\gamma_{\Delta(T)}, +\infty]$

Table 3.1: Classification of simplices in the λ-complex. (Top rows) Common classification with α-complex. (Bottom rows) λ-complex specific cases.

3.3.6 Tracking Topological Events

Consider the space-filling diagram $\mathscr{F}_\lambda$. Selected values of λ featured in Table 3.1 correspond to topological events underwent by $\mathscr{F}_\lambda$—in terms of homology groups. Of particular interest for the application sketched in section 5.2.2 are those events triggering a decrease of the number of connected components of $\mathscr{F}_\lambda$. Such events are associated with selected one-dimensional simplices of the dual complex, and the connected components can be maintained by a Union-Find algorithm upon sorting the λ values featured in Table 3.1. Following classical terminology, the lifetime of a c.c. is called its *topological persistence* [ELZ02, CSEH05].

3.4 Algorithms

In this section, we present an output sensitive algorithm to compute the dual complex, together with the accompanying algorithms to compute the λ-complex and a variant which we call the *reduced* λ-complex.

3.4.1 Using a Sentinel Ball

To ease the implementation, denoting $\mathscr{T}$ the input balls and $\overline{\mathscr{T}}_{\max}$ the maximal balls of $\mathscr{T}$, we define a new ball $\overline{S}_{\max}$ which is the only maximal ball in $\mathscr{S} = \mathscr{T} \cup \{\overline{S}_{\max}\}$, and we compute the CW diagram of $\mathscr{S}$. Note that the neighbors of $\overline{S}_{\max}$ in the CW diagram of $\mathscr{S}$ are the toleranced balls of $\mathscr{T}$ bounding the CW convex hull of $\mathscr{T}$. To define $\overline{S}_{\max}$, we successively set its extremal radii and its center $c_{\overline{S}_{\max}}$.

First, the extremal radii are chosen such that $\overline{S}_{\max}$ is maximal. We arbitrarily set $r^-_{\overline{S}_{\max}} = 0$, and set $r^+_{\overline{S}_{\max}}$ so that $\overline{S}_{\max}$ is maximal, that is:

$$r^+_{\overline{S}_{\max}} = 1 + \max_{\overline{B_i} \in \overline{\mathscr{T}}_{\max}} (\delta_i). \tag{3.36}$$

To set the center, we first compute the radius λ' of the largest extremal TT ball to all tuples (pairs, triples, quadruples) of toleranced balls in $\mathscr{T}$. (Because of domination, we need to process not only quadruples but also pairs and triples.) Consider now a toleranced ball $\overline{B_i} \in \mathscr{T}$. Center $c_{\overline{S}_{\max}}$ is chosen such that the radius of the smallest TT ball of the pair $(\overline{B_i}, \overline{S}_{\max})$ is larger than $\lambda', \forall i$. (It is in fact sufficient to process toleranced balls which are maximal in $\mathscr{T}$.) From Eq. (3.18) with $\alpha = 1$ and $\beta = 1$, this condition reads as:

$$|| \ c_i - c_{\overline{S}_{\max}} \ || \geq \lambda'(\delta_i + \delta_{\overline{S}_{\max}}) + (r^-_{\overline{S}_{\max}} + r^-_i). \tag{3.37}$$

Without loss of generality, we choose $c_{\overline{S}_{\max}}$ on the z-axis. Since $\lambda'\delta_i + r^-_i \geq 0$ for any toleranced ball $\overline{B_i} \in \mathscr{T}$, squaring Eq. (3.37) yields the following equivalent degree two condition:

$$f(z_{max}) = || \ c_i - (0,0,z_{max}) \ ||^2 - (\lambda(\delta_i + \delta_{\overline{S}_{\max}}) + (r^-_{\overline{S}_{\max}} + r^-_i))^2 \geq 0 \tag{3.38}$$

Function $f(z_{max})$ is always positive, or is so for two intervals $(-\infty, z^-_i)$ and $(z^+_i, +\infty)$ with $z^-_i \leq z^+_i$. It is therefore sufficient to set $z_{max} > z^+_i, \forall i$.

3.4.2 Hasse Diagrams of Tuples and Simplices

Tuples

A tuple reduces to the list of indices of the toleranced balls it contains, and the inclusion between these indices defines a partial order. We shall use it to store selected tuples called candidate tuples into a Hasse diagram denoted D_T, see next section.

Simplices

We represent the dual complex by the Hasse diagram D_S introduced in section 3.3.2. The level $D_S(k)$ features the simplices of dimension k, and the predecessors and successors of a node in D_S respectively represent the faces and cofaces of the corresponding simplex. A node with no successor is called *terminal*. For two consecutive levels $D_S(k)$ and $D_S(k+1)$, the *slice graph* $D^{Sl}_S(k,k+1)$ is defined as follows : the nodes of $D^{Sl}_S(k,k+1)$ are those of $D_S(k)$; two such nodes are incident if they share a coface of $D_S(k+1)$.

Hasse Diagram D_S and Related Operations

We endow D_S with two operations to be used for the construction of the dual complex. Consider a tuple T_k. This tuple is said to *identify* a $(k-1)$-face if the vertices defining this $(k-1)$-face correspond to the toleranced balls of the tuple. Finally, consider the graph $D^{Sl}_S(k,k+1)$, together with a k-simplex $\Delta(T_{k+1})$ having a $(k-1)$-face identified by T_k, that is $T_{k+1} = T_k \cup \{\overline{B_j}\}$ for $\overline{B_j} \in \overline{\mathscr{S}} \backslash T_k$. In order to deal with non-simply connectedness and non connectedness of Voronoi faces, we also define a *restrained connected component* (restrained c.c.) of $D^{Sl}_S(k,k+1)$ anchored at $\Delta(T_{k+1})$, as a maximal c.c. such that all its nodes are identified by the tuple T_k. Consider now a maximal set of restrained c.c. of $D_S^{Sl}(k,k+1)$ connected by nodes of $D_S^{Sl}(k,k+1)$ which are not identified by the tuple T_k. Such a set defines an *unrestrained c.c.*. These notions are illustrated on Figure 3.14.

Remark 12. *For a non connected Voronoi region, an unrestrained c.c. may contain one ore several restrained c.c.:*
– if the unrestrained c.c. matches a restrained c.c., the dual of the nodes of this c.c. bound a hole. This is the case of C_2 on Fig. 3.14.
– if the unrestrained c.c. contains several restrained c.c.: the dual of the nodes of the restrained c.c. do not bound a hole; furthermore, each restrained c.c. is associated to a different Voronoi face of the Voronoi region. This is the case of C_1 and C_3 on Fig. 3.14.

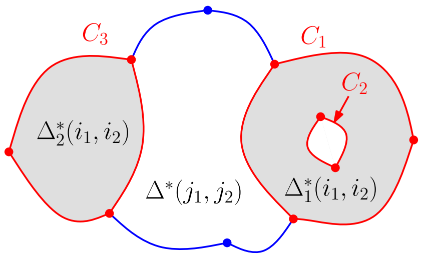


Figure 3.14: Computing Voronoi faces using restrained and unrestrained connected components. Example of two toleranced balls $\overline{B_{i_1}}$ and $\overline{B_{i_2}}$ whose Voronoi region consists of Voronoi faces $\Delta_1(i_1,i_2)^*$ and $\Delta_2(i_1,i_2)^*$ respectively bounded by two and one cycle (C_1, C_2 and C_3). Color codes for Voronoi edges and vertices : red: Voronoi faces bounding the Voronoi region of $(\overline{B_{i_1}}, \overline{B_{i_2}})$.; blue: Voronoi faces not bounding the Voronoi region of $(\overline{B_{i_1}}, \overline{B_{i_2}})$. Each cycle C_1, C_2 and C_3 is a restrained connected component (c.c.). There are two unrestrained c.c., one grouping C_1 and C_3 (these restrained c.c. respectively correspond to the connected region containing the Voronoi face $\Delta_1(i_1,i_2)^*$, and to $\Delta_2(i_1,i_2)^*$), and one containing only C_2 (corresponding to the hole in $\Delta_1(i_1,i_2)^*$). See Remark 12.

Mapping Tuples to Simplices

As seen in section 3.3.2, a tuple T possibly yields several simplices. Denoting m the multiplicity of a tuple T_k, we shall use a map M_{TS} mapping T to the corresponding simplices $\Delta_j(T), j \in 1,\dots,m$. The correspondence between the levels of D_S and D_T is illustrated on Figure 3.15. Abusing terminology, we define:

Definition. 4. *A simplex of the dual complex is said to be* identified *by a tuple T if the toleranced balls found in T form a subset of the vertices of the simplex. The* cofaces *of the tuple T are the simplices of the dual complex which are identified by T.*

Map M_{TS} is used to retrieve the cofaces of a tuple T as follows: first, the successors of T in the Hasse diagram D_T are collected; second, the simplices associated to each successor are accessed thanks to the map M_{TS}.

$$\begin{tikzcd}
\text{Layers of Hasse diagrams} & \\
k+1\text{-tuples} & k\text{-simplices} \\
D_T(4) \arrow[r] & D_S(3) \\
D_T(3) \arrow[r] & D_S(2) \\
D_T(2) \arrow[r] & D_S(1) \\
D_T(1) \arrow[r] & D_S(0) \\
\text{map } M_{TS} &
\end{tikzcd}$$

Figure 3.15: Correspondence between the layers in the Hasse diagram of tuples D_T and the Hasse diagram of simplices of the dual complex D_S.

3.4.3 Computing Candidate Tuples

A number of practical settings are concerned with growth processes up to a maximum value $\lambda_{\max} < \Lambda_{\max}$, although in the absence of restriction we shall use $\lambda_{\max} = +\infty$. We now define so-called candidate tuples, which will be used in section 3.4.4.

Definition. 5. *A k-tuple with $k \leq 2$ is termed a* candidate tuple *if its minimal TT ball has a radius less than $\lambda_{\max}$. Similarly, a 4-tuple is called* candidate *if at least one radius of its TT balls is less than $\lambda_{\max}$.*

We report candidate tuples, from singletons to quadruples. The strategy consists of building the Hasse diagram D_T in a bottom-up fashion, the last two layers being constructed from the layers below. More precisely:

▷ A toleranced ball is a candidate singleton provided that $-r_i^-/\delta_i > \lambda_{\max}$. If so, we store it into L_1.

▷ Consider a pair $(\overline{B_i}, \overline{B_j})$ out of the $\binom{n}{2}$ possibly pairs. The pair is a candidate provided that $\overline{B_i}[\lambda_{\max}] \cap \overline{B_j}[\lambda_{\max}] \neq \emptyset$. If so, we store it into L_2 and set the links to L_1.

▷ For triples and quadruples, we exploit the recursive structure of tuples encoded in the Hasse diagram D_T. Denote L_k the list of candidate k-tuples. We wish to compute L_{k+1} from the lists $L_i, i = 0, \ldots, k$ and D_T. Let a be a node from L_{k-1}. For two nodes c and d which are successors of a in the Hasse diagram, one has $\mid a \cup b \mid = k+1$. That is, all candidate $(k+1)$-tuples can be formed by examining all pairs of successors of nodes in L_{k-1}. We also set the diagram D_T along the way.

Using this strategy yields the following:

Observation. 1. *Denote n the number of toleranced balls and τ' the number of candidates tuples. Computing all candidate tuples has output sensitive complexity $O(n^2 + \tau')$. Moreover, checking that the associated extremal TT balls are conflict free has complexity $O(n(n^2 + \tau'))$.*

Proof. The quadratic term comes from the possible $\binom{n}{2}$ pairs. For triples and quadruples, it is sufficient to observe that a k-tuple associated to a $(k-1)$-simplex is discovered a number of times equal to the number of its $(k-3)$-faces, that is, a triple is discovered three times and a quadruple six times. Hence the output sensitive complexity for triples and quadruples.
For the second part of the claim, observe that for each candidate tuple, one needs to run one (four) iterations on all remaining balls for singletons/pairs/triples (quadruples). □

3.4.4 Top-down Construction of the Dual Complex

To build the dual complex, assume that the pre-processing described in section 3.4.3 has been carried out, with $\lambda_{\max} = +\infty$. The algorithm builds the CW VD from cells to vertices. Three data structures are manipulated. First, the Hasse diagram D_T is used and updated, since some candidate simplices which do not yield simplices are removed. Second, the Hasse diagram D_S is constructed. Finally, the map M_{TS} mapping tuples to simplices is used and incrementally updated.

Computing $D_S(3)$ or equivalently the 0-skeleton of the CW VD

We examine each candidate 4-tuple, and create one simplex in $D_S(3)$ for each conflict-free solution of system (3.30). Map M_{TS} is set accordingly.

Computing $D_S(2)$ or equivalently the 1-skeleton of the CW VD

Case analysis. This step consists of computing Voronoi edges corresponding to $D_S(2)$ and connecting them to Voronoi vertices associated with $D_S(3)$. We consider the candidate 3-tuples. Each such tuple possibly contributes one or more Voronoi edges, and we face three cases. Assume that the cofaces of the tuple have been collected thanks to map M_{TS}.

▷ **Vor-1a** If the tuple does not have any coface in $D_S(3)$ and its maximal TT ball has a conflict, the simplex does not exist in the dual.

▷ **Vor-1b** If the tuple does not have any coface in $D_S(3)$ and has a conflict-free maximal TT ball, the simplex is dominant and contributes its full bisector as a Voronoi edge.
▷ **Vor-1c** If the simplex has cofaces in $D_S(3)$, assuming that $\overline{\mathscr{S}}_{\max}$ contains a single ball, it contributes Voronoi edges bounded by Voronoi vertices. This number of vertices is even, and the construction of Voronoi edges is a two-stage process. First we sort the vertices along the bisector. To do so, we process separately the vertices found in the two half-spaces delimited by the plane containing the centers of the toleranced balls. Sorting either set of Voronoi vertices along the bisector is tantamount to sorting the weights of the TT balls associated with these Voronoi vertices. Second, we form the curved Voronoi edges. If the smallest TT ball of the tuple is conflict-free, there is a Voronoi edge between the two first Voronoi vertices of each half-space, and this edge determines the remaining Voronoi edges in each half-space. Otherwise, there is a Voronoi edge between the first two Voronoi vertices on each side of the plane – if any.

Algorithms. We examine the cases in turn.
▷ **Vor-1a** The sterile tuple is removed from D_T.
▷ **Vor-1b** The simplex is created, and the data structures D_S and M_{TS} are updated accordingly.
▷ **Vor-1c** Sorting Voronoi vertices along a bisector requires two predicates: the `Orientation` predicate to locate the vertices in the two half-spaces; a comparison of roots of degree four polynomials to compare the radii of extremal TT balls.
In terms of data structures, every simplex created triggers an update of D_S and M_{TS}.

Computing $D_S(1)$ or equivalently the 2-skeleton of the CW VD

Case analysis. This step consists of computing Voronoi 2-faces corresponding to $D_S(1)$ and connecting them to Voronoi edges associated with $D_S(2)$. Building a Voronoi 2-face requires identifying all the bounding Voronoi vertices i.e. Voronoi 0-faces, which are glued together by Voronoi 1-faces. As illustrated on Figure 3.14, this is a non trivial task since the Voronoi region of a pair might not be connected, and a face might not be simply connected. Let the *support* of a non simply connected face be the simply connected region which contains it. We consider all the candidate pairs, and for each of them analyze the cofaces collected thanks to map M_{TS}. We face three cases.

▷ **Vor-2a and Vor-2b** These two cases are similar to those found for triples : a pair which does not have any coface is either dominant or is not present in the dual complex.
▷ **Vor-2c** The third case is the complex one. Using the 1-skeleton of the CW-VD, we identify those cycles bounding the supports, and those bounding holes. To do so, we search the restrained and unrestrained c.c. of $D_S^{SI}(2,3,)$. There are three sub-cases:
– *Vor-2c-1.* The Voronoi region of the 2-tuple is simply connected : there is one restrained and one unrestrained c.c. which are identical.
– *Vor-2c-2.* The Voronoi region is not connected (the topology of the faces are arbitrary): restrained and unrestrained c.c. differ. An unrestrained c.c. consists of the union of one or more restrained c.c.. If there is only one restrained c.c. in an unrestrained c.c., the cycle is one hole. If not, the cycles bound faces. As an example, consider Figure 3.14. Each cycle C_1, C_2, C_3 is a restrained c.c.. There are two unrestrained c.c.: one includes cycles C_1 and C_3 which are connected in the 1-skeleton of the CW VD, the other one is C_2 which is not connected to C_1 and C_3 in the 1-skeleton of the CW VD.
– *Vor-2c-3.* The Voronoi region is connected but not simply connected. The two searches yield several c.c. which are the same for the restrained and unrestrained case.

Algorithms. We focus on the complex case. Let $T = (\overline{B_i}, \overline{B_j})$ be the pair being processed.
▷ **Vor-2c** From an algorithmic standpoint, the computation of restrained and unrestrained c.c. is a two-stage process. First, the (plain) c.c. of graph $D_S^{SI}(2,3)$ are computed. Any c.c. containing nodes identified by the pair T is an unrestrained c.c.. For example, on Figure 3.14, the process yields two such c.c., namely the c.c. defined by C_2, and that involving C_1, C_3 and the blue edges.
Second, we run a union-find algorithm on each such c.c.. More precisely, consider the subset $D_S(3 \mid T)$, that is the nodes of $D_S(3)$ which are identified by the pair T. These nodes correspond to Voronoi vertices involving the two balls. Similarly, consider the subset $D_S(2 \mid T)$ of nodes of $D_S(2)$ which are identified by the pair T. These nodes

of D_S correspond to Voronoi edges involving the two balls. We run a union-find process with node set $D_S(2 \mid T)$ and edge set $D_S(3 \mid T)$. As illustrated on Figure 3.14, this process yields the restrained c.c..

In terms of data structures, the creation of a simplex triggers an update of D_S and M_{TS}.

Remark 13. *For case* Vor-2c-2*, we do not know which faces of the Voronoi region bound cycles defining holes. However, this information is irrelevant if we only focus on the neighborhood relationship between Voronoi regions.*

Remark 14. *For case* Vor-2c-3*, one can further identify the cycle bounding the support. Let $\overline{B_i}$ and $\overline{B_j}$ be the toleranced balls of the pair. Consider a cycle C, and let δ_C be the maximum δ of the toleranced balls involved in Voronoi edges and vertices along C—and different from $\overline{B_i}$ and $\overline{B_j}$. If $\delta_C < \min\{\delta_i, \delta_j\}$, then cycle C bounds a hole, and reciprocally. To see why, consider the bisector of the toleranced balls associated to δ_C and $\min\{\delta_i, \delta_j\}$: it bounds the Voronoi region of the toleranced ball associated to δ_C iff $\delta_C < \min\{\delta_i, \delta_j\}$.*

Computing $D_S(0)$ or equivalently the 3-skeleton of the CW VD

Case analysis. This step consists of computing Voronoi 3-faces corresponding to $D_S(0)$ and connecting them to Voronoi 2-faces associated with $D_S(1)$. Building a Voronoi cell requires identifying all the bounding Voronoi 2-faces, which are glued together by Voronoi edges, i.e. Voronoi 1-faces. To do so, the difficulties are identical to those faced to compute the 2-skeleton since the topological complications are the same—non connectedness and non-simply connectedness. Analyzing the cofaces found for each toleranced ball yields the following two cases.

▷ **Vor-3a** If a toleranced ball has no coface, its Voronoi region is empty.

▷ **Vor-3b** If a toleranced ball has at least one coface, we use the algorithm computing $D_S(1)$ using $D_S^{Sl}(1,2)$ instead of $D_S^{Sl}(2,3)$. Note that if two dual triangles have a common bounded dual tetrahedron, they share at least one dual edge.

Algorithms. Let T be the tuple of interest, which consists of one ball. To glue Voronoi 2-faces thanks to Voronoi 1-faces, union-find is run on the node set $D_S(1 \mid T)$ and edge set $D_S(2 \mid T)$.

Complexity analysis

Denote `Sorting`(A) the cost of sorting set A, and `Union-find`(A,B) the cost of running a union-find algorithm on node set A using the edge set B.

The following observations, which directly stem from the description of algorithms, show that the algorithm constructing the dual complex has output sensitive complexity:

- Computing the 1-skeleton has complexity $\sum_{T \in G_T(3|T)} \texttt{Sorting}(D_S(3 \mid T))$
- Computing the 2-skeleton has complexity $\sum_{T \in G_T(2|T)} \texttt{Union-find}(D_S(3 \mid T), D_S(2 \mid T))$
- Computing the 3-skeleton has complexity $\sum_{T \in G_T(1|T)} \texttt{Union-find}(D_S(2 \mid T), D_S(1 \mid T))$

Analyzing these complexities is directly related to the complexity of the CW diagram, an open problem to the best of our knowledge.

It should be noticed, though, that the cubic pre-processing might be optimal in the worst-case. Indeed, the worst-case complexity of the diagram is clearly at least quadratic. And since a Voronoi region can be disconnected, incremental algorithms aiming at finding conflicts may have to exhaustively probe the whole diagram.

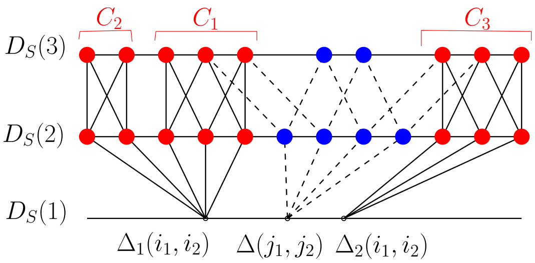


Figure 3.16: Computing dual simplices. Hasse diagram representation of dual complex of example of Figure 3.14. Color of simplices are the same that color of their dual in Figure 3.14. Links between simplices whose duals bound the Voronoi faces of the pair $(\overline{B_{i_1}}, \overline{B_{i_2}})$ are represented in solid lines.

3.4.5 Computing the (reduced) λ-complex

Representation

In the λ-complex, a simplex Δ is attached three tags stating whether (i) it is Gabriel or not, (ii) it contributes to the convex hull $CH(\overline{\mathscr{S}})$ or not, and (iii) it is dominant, dominated, or neither one nor the other. Moreover, Δ is endowed with three values delimiting the intervals of a row in Table 3.1.

Computation

The classical way to compute interval for simplices in the affine α-complex consists of visiting simplices in a top-down fashion, namely from tetrahedra to vertices [Ede92]. In doing so, the status and intervals of a simplex are inferred from those of its cofaces. We apply this strategy for terminal nodes in the Hasse diagram, which are either tetrahedra in $D_S(3)$ or selected dominant nodes of $D_S(2)$ and $D_S(1)$.

On the reduced λ-complex

Consider the case where one wishes to explore the growth process of the toleranced balls up to a maximum value $\lambda_{\max} < \Lambda_{\max}$ of λ. We call the collection of simplices that appear in the λ-complex for $\lambda \leq \lambda_{\max}$ the *reduced* λ-complex. Computing the reduced λ-complex requires processing a subset of all tuples involved in the entire λ-complex.
Having computed the candidate tuples as indicated in section 3.4.3, the computation of the reduced λ-complex is identical to that of the entire λ-complex.

3.4.6 Implementation

Sketch

The implementation follows the CGAL spirit, see `http://www.cgal.org`, and we sketch it in terms of *concepts* (a set of requirements) and *models* (a particular implementation). The class `CW_dual` representing the dual complex is templated by a combinatorial class providing the Hasse diagram representation, and by a geometric concept class `CWGeometricKernal` providing the predicates and constructions required. The corresponding generic model `CW_geometric_kernel` is itself templated by a concept class `AlgebraicKernel` providing the operations needed to deal with the extremal TT balls. As specified by propositions 2 and 3, computing these TT balls requires solving linear systems or a degree four algebraic equation, while the conflict-free test requires evaluating the conflict_free predicate of Eq. (3.10). We implemented a model of the `AlgebraicKernel` named `CW_algebraic_kernel_double` which uses CGAL's `Algebraic_kernel_d_1`—the latter provides efficient operations on univariate polynomials. The number type being `double`, this kernel does not provide exact predicates. Finally, the class `CW_alpha_shape` inherits from `CW_dual` and provides the tags and intervals detailed in Table 3.1.

Sanity Check

To probe the implementation, given a collection of toleranced balls with identical parameters, we checked its ability to compute the Delaunay triangulation of the centers.

Performance

To scale the implementation, we ran it on random collections up to 1000 toleranced balls on a DELL computer with Intel Xeon processor at 3.2 GHz with 2048 Mo of RAM. Balls were generated as follows: the set $\mathscr{C}$ of centers is uniformly generated at random in a cube; for each center c_i, radius r_i^- is set to the length of the shortest edge between c_i and its neighbors in the periodic Delaunay triangulation of $\mathscr{C}$, while r_i^+ is set to the mean between r_i^- and the length of the longest edge between c_i and its neighbors in the periodic Delaunay triangulation of $\mathscr{C}$ [CT09]. An example with 200 toleranced balls is shown on Figure 3.17.

Statistics for the reduced Dual Complex computation up to $\lambda = 1$ are reported on Figure 3.18. We note in particular that the number of candidate tuples increases linearly with the number of toleranced balls, as a linear regression gives a slope of 515 with R-squared value of 0.99. So does the running time, which is about 251 minutes for 1000 toleranced balls.

An example calculation for the whole dual complex of 200 random toleranced balls is illustrated on Figure 3.19, with the distribution of λ values associated to Gabriel simplices *and* tetrahedra.(Note that in the affine case, all tetrahedra are Gabriel, a property which does not hold in our case since four balls may contribute four tetrahedra — one of them may be Gabriel.) The whole calculation took about 69 hours for $38,515,103$ candidate tuples and 5004 simplices. There are 1971 Gabriel simplices and 1148 tetrahedra. Note that $88,8\%$ of Gabriel simplices and tetrahedra appear in the λ-complex for $\lambda \leq 1$.

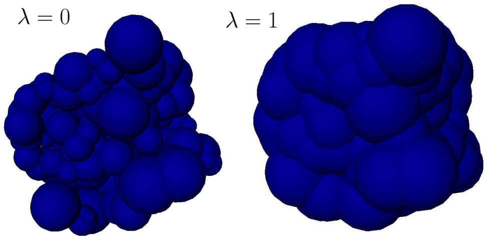


Figure 3.17: Example of 200 toleranced balls uniformly generated at random in a cube. **Right.** Inner balls ($\lambda = 0$). **Left.** Outer balls ($\lambda = 1$).

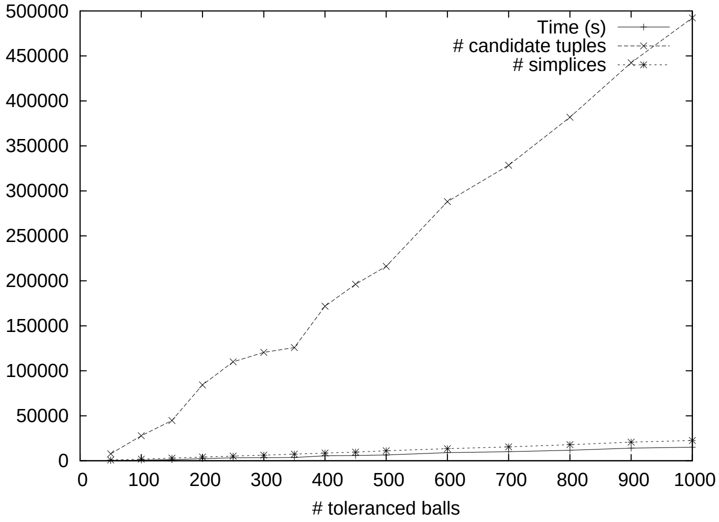


Figure 3.18: Statistics for the reduced λ-complex up to $\lambda = 1$ for a random configuration of 200 toleranced balls.

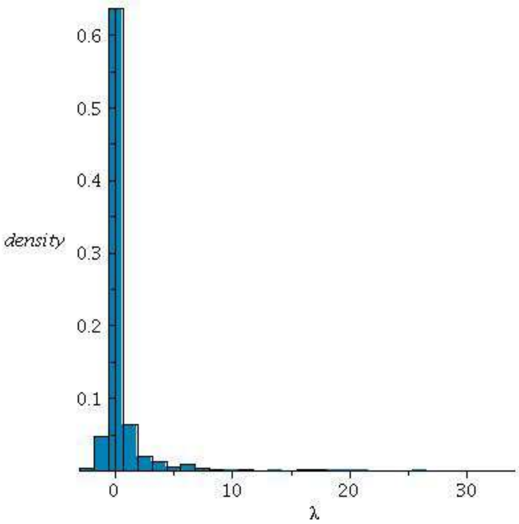


Figure 3.19: Distribution of λ values associated with Gabriel simplices and tetrahedra for a random configuration of 200 toleranced balls.

Chapter 4

The Nuclear Pore Complex: Material and Methods

4.1 Introduction - Rationale

As shown in Chapter 1, the reconstruction of the NPC in [ADV$^+$07a] is qualitative and analyzing directly the information contained in probability density maps is a hard task. Our solution to gain this understanding consists of using toleranced models, built from these probability density maps. In this perspective, the following elements are presented in this chapter.

First, we discuss precisely several sub-complex of the NPC, namely the Nup84 sub-complex (called Y-complex), the Nic96 sub-complex (called T-complex) and the Nup82 sub-complex. In doing so, we present so-called *skeleton graphs*, which are graphs encoding the pairwise protein contacts within these these complexes. In Chapter 6, these graphs will be used as probes to challenge the complexes observed in our toleranced models.

Second, we present in Section 4.3 a number of analysis on the density maps underlying our toleranced models. Assessing the uncertainties of these maps is indeed mandatory to understand selected features of our toleranced models.

Finally, in Section 4.4, we present the algorithm used to construct toleranced models of the whole NPC from the density maps.

4.2 Sub-systems of Interest

4.2.1 The Y-complex and Related Complexes

Each half spoke of the NPC restricted to the coat cylinder contains a heptamer called the Y-complex (Nup133, Nup84, Nup145C, Sec13, Nup120, Nup85 and Seh1). To describe two models of the Y-complex and its embedding in the NPC, illustrated on Figure 4.1, we decompose the Y-complex into sub-systems, namely [1] the Y_X-short-arm, the Y_X-long-arm, the Y_X-edge, the Y_X-tail, and the Y-arms, the Y-core, the Y-main, and Y-junction. The model of the Y-complex proposed by Blobel et al. [KB09] presented on Fig. 4.1(Top left) comes from a reconstruction involving single particle EM data, together with crystal structures of the Y_X-short-arm, the Y_X-long-arm, the Y_X-edge and Nup133. Using size-exclusion chromatography and analytical ultracentrifugation, the authors show that two opposite proteins of the Y-complex namely (Nup120, Nup133) interact in a head-to-tail fashion [SMD$^+$09], a contact motivating the embedding of copies of the Y-complex into the NPC in a ring-like fashion. Kampman et al [M. 11] recently provided a strong support for an head to tail arrangement of these complexes, in which 8 Y-complexes lie with their long axes parallel to the nuclear envelop plane and form two rings through interaction of Nup133 with the arms of the neighboring Y-complexes.
Using the same pairwise contacts together with those involved in the Y_X-tail, Brohawn et al. [BS09] propose an embedding of the Y-complex into the NPC where the Y_X-tail extremities point towards the cytoplasmic and

[1] The subscript X in Y_X hints at a crystal structure.

nuclear hemispheres of their respective half-spokes. This proposal is motivated by homology considerations with coat vesicles and interactions with T-complexes. We shall investigate these models using the skeleton graphs of Figure 4.1.

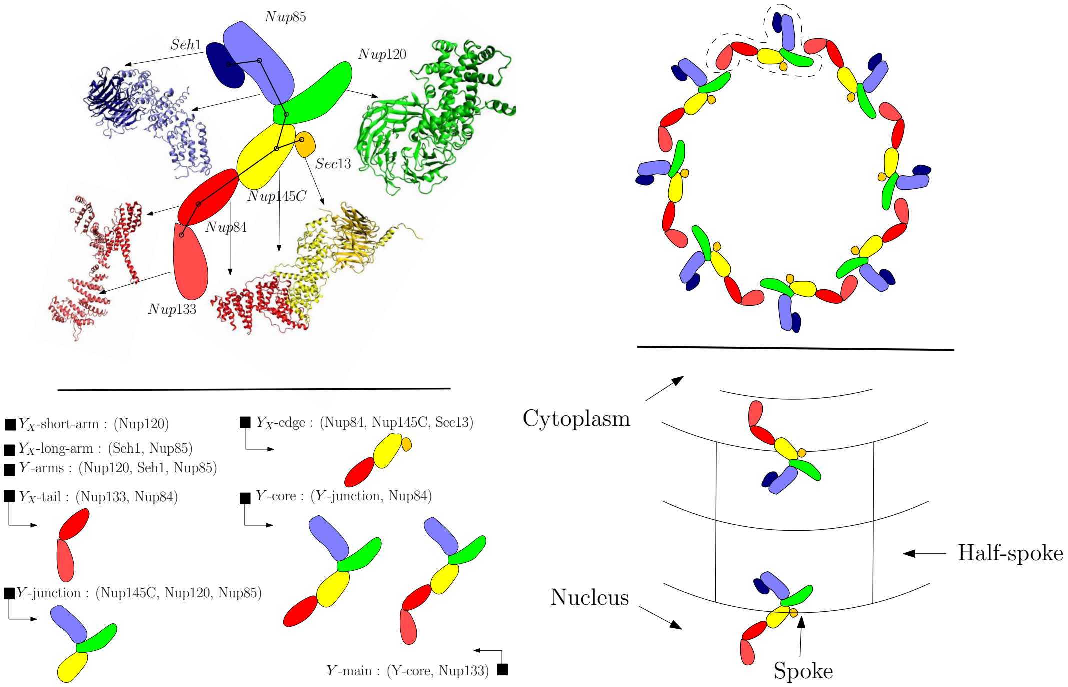

$Nup85$
$Seh1$
$Nup120$
$Sec13$
$Nup145C$
$Nup84$
$Nup133$
■ Y_X-short-arm : (Nup120)
■ Y_X-long-arm : (Seh1, Nup85)
■ Y-arms : (Nup120, Seh1, Nup85)
■ Y_X-tail : (Nup133, Nup84)
■ Y-junction : (Nup145C, Nup120, Nup85)
■ Y_X-edge : (Nup84, Nup145C, Sec13)
■ Y-core : (Y-junction, Nup84)
Y-main : (Y-core, Nup133) ■
Cytoplasm
Half-spoke
Nucleus
Spoke

Figure 4.1: Model of the Y-complex and its embedding in the NPC. (**Top Left.**) Putative Y-complex model after [KB09]. It is composed of 7 proteins: Nup133 (light red), Nup84 (red), Nup145C (yellow), Sec13 (orange), Nup120 (green), Nup85 (blue) and Seh1 (dark blue). The skeleton graph $G_t(Y)$ of the Y-complex is represented in black solid lines. **(Bottom Left.)** Terminology used for sub-complexes of the Y-complex. **(Top right.)** Putative arrangement of Y-complexes, from [KB09]. Interactions between Nup133 and Nup120 account for one ring of head-to-tail Y-complexes in the cytoplasmic and nuclear hemispheres. **(Bottom right.)** Putative arrangement of Y-complexes in a spoke [BS09]. Each spoke of the NPC contains two Y-complexes with Y_X-tail pointing towards the cytoplasmic and nuclear hemispheres.

4.2.2 The T-complex and Related Complexes

The nucleoporin Nic96 is located in the adapter cylinder, and makes the T-complex with instances of Nsp1, Nup49 and Nup57 located in the channel cylinder. We split these proteins into the T-core i.e. (Nic-96, Nsp1) and the T-leg i.e. (Nup49, Nup57), see Figure 4.2. Filaments of the latter three proteins are involved in the regulation of the traffic trough the NPC. We are not aware of any crystal structure of complexes involving two or more such proteins.

Contacts between proteins of the T-core were determined by purification experiments [GDH93]. Similarly, it has been shown that Nup57 binds Nsp1 and Nup49 independently [SHL$^+$97], which motivates the first skeleton graph $G_t(T)$ of the T-complex. A second skeleton graph $G_t(T\text{-comp})$ encodes all possible contacts. This model is warranted by in vitro binding assay experiments [SSF$^+$08] showing interactions between the filaments of the T-leg proteins with Nic96. Finally, motivated by the graph analysis presented in Chapter 6, we introduce the skeleton graph $G_t(T\text{-new})$ referring to $G_t(T\text{-comp})$ without the contact between Nup57 and Nic96. See Figure 4.2.

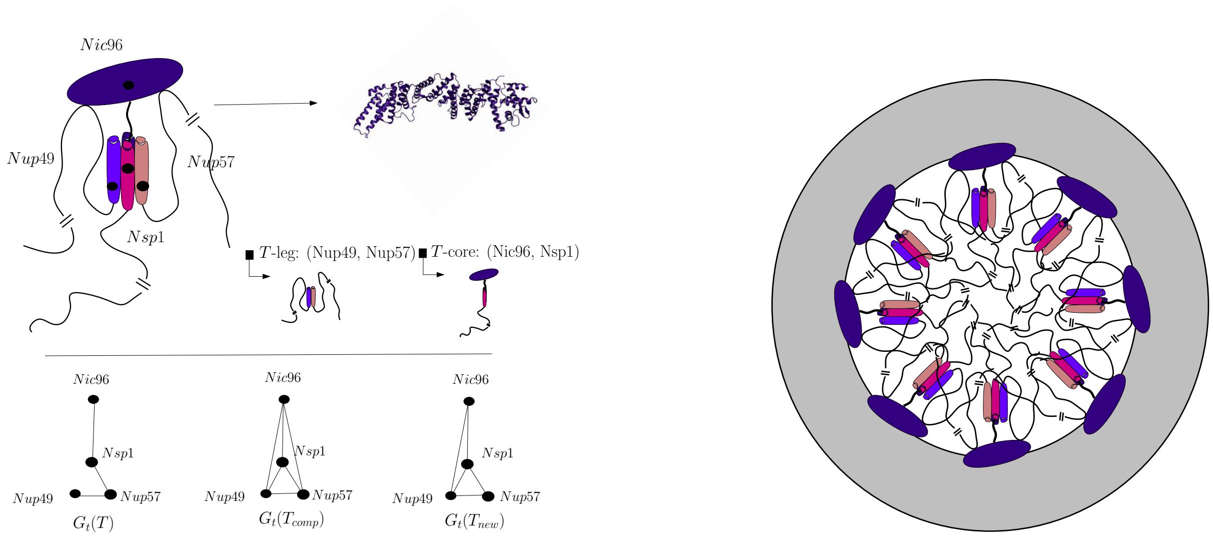


Figure 4.2: Model of the T-complex and its embedding in the NPC **(Top Left.)** The T-complex consists of Nic96 (dark blue), Nsp1 (magenta), Nup49 (light blue) and Nup57 (apricot). Filaments are non structured domains of Nsp1, Nup49 and Nup57. **(Bottom Left.)** The skeleton graphs for the T-complex. **(Right.)** The putative location of instances of the T-complex in the inner rim of the NPC.

4.2.3 The Nup82 Sub-Complex and Related Complexes

Restrained to the channel cylinder, the Nup82-complex is a trimeric complex involving Nup82, Nsp1 and Nup159, especially localized in the cytoplasmic side [BSHP$^+$98]. Interactions between Nup82 and Nsp1 was shown by Eduard C. Hurt et al [GEW$^+$95] using affinity purification with tagged Nsp1. However, mutant analysis and affinity purification of tagged Nup82 reveal that the instances of Nsp1 involving in the T-complex are not the ones interacting with Nup82. Interactions between Nup82 and Nup159 were shown by Blobel et al [HdcB98]. Interestingly, transcriptional repression experiments show that Nup82 functionally interacts with the RNA transport through the NPC, and Nup159 has a FXFG repeat sequence involving in the transport of mRNA through the NPC.

4.3 On the Density Maps Used

The quality of the toleranced models built in section 4.4 depends on the accuracy of the probability density maps used. All the probability density maps are shown in supplemental Section 4.5.1. A sketchy model of a cytoplasmic half-spoke of the NPC derived from the probability density maps is shown in Figure 4.3. It is meant to intuitively position the proteins of the NPC with respect to one another.

In the following, on a per-density map basis, we report statistics aiming at qualifying these maps, in particular regarding the number of connected components (c.c.) of voxels having a non null probability in Section 4.3.1, and the volume of these c.c. with respect to their expected volume in Section 4.3.2.

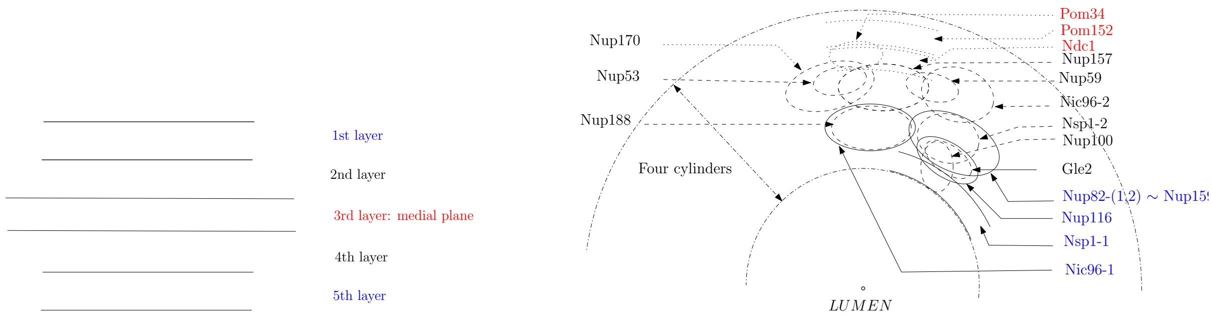


Figure 4.3: Sketchy model for the architecture of the NPC. **(Left)** Side view **(Right)** Top view of a half-spoke from the cytoplasmic side. Proteins are arranged into 5 layers, the middle one corresponding to the medial plane. Proteins of the first three layers only are represented. Note that Nup145N is the only protein present on the nuclear side only.

4.3.1 On the Number of Connected Components

Ideally, the number of c.c. of a map should match the stoichiometry of the corresponding protein. But this is not always the case. To further this observation, Top of Figure 4.4 displays the number of c.c. for each density map: this number is larger than / equal to / less than the stoichiometry in five / 19 / nine cases.

The 19 cases for which the stoichiometry matches with the number of c.c. are: Pom34, Seh1, Nup49, Nup57, Nup145C, Nup82-1, Nup82-2, Nup84, Nup85, Nsp1-2, Nic96-1, Nic96-2, Nup100, Nup1, Nup120, Nup133, Nup159, Nup170 and Nup188.
The nine cases for which the stoichiometry is lower than the number of c.c. are: Sec13, Gle2, Nup42, Nup53, Nup59, Nup60, Nup145N-1, Nup145N-2 and Nup116. They correspond to ambiguous locations which induce multiple connected components per instance. However, except for Sec13 and Nup116, the c.c. of the seven remaining density maps may be visually grouped in the expected number of distinct clusters, avoiding ambiguity on the location of instances.
The five cases for which the stoichiometry is larger than the number of c.c. are: Ndc1, Nsp1-1, Pom152, Nup157 and Nup192. These cases occur when multiple c.c. located nearby merge. For Nsp1 and Ndc1, c.c. that are on the same side of the NPC, but in two different spokes merge. For Nup170 and Nup192, two c.c. on the same spoke but on both sides of the NPC merge. This phenomenon is extreme for Pom152, since a single c.c. corresponding to a filled torus is observed. Note that merging the two density maps of Nup82 (called Nup82-1 and Nup82-2) results in eight c.c., see Figure 4.19. However, each density map taken separately has eight c.c., allowing one to assign one c.c. to each instance of Nup82.

4.3.2 On the Volume of Connected Components

In this analysis, we restrict ourselves to maps which have the correct stoichiometry, since the meaning of c.c. in the remaining cases is unclear. For example, a c.c. within a plethoric map could be significant or could be insignificant, depending on the values of the probabilities observed in this c.c.. In theory, analysing the relative importance of c.c. in any map can be done using Morse theory and persistence theory, in a manner similar to the algorithms developed in [CCS11] in the context of Morse theory of the distance function. Yet, for general (density) maps, effective algorithms for Morse-Smale decompositions yet have to be developed.
Let P be a protein type. We call *reference volume* $Vol_{ref}(P)$ an estimation of the volume of P from its sequence [HGC94]. Assume that the density map of P contains n c.c.. Denoting $Vol(cc_i)$ the volume of the ith c.c., consider the set of volume ratios

$$v_{cc_i} = Vol(cc_i)/Vol_{ref}(P), \text{ for } i = 1, \ldots, n. \tag{4.1}$$

We have drawn the box plots [2] of the 19 density maps with correct stoichiometry on Bottom of Figure 4.4. Second, the whiskers extend to the extrema values of the plot, limited by 1.5 times the inter-quartile distance.

[2] Recall that the box plot of a set of values is presented as follows. First, the rectangle displays three values, namely the first and third quartiles (small sides of the rectangle), and the median (bold line-segment inside the rectangle).

Values below and above these thresholds are represented by circles. While most of the proteins have median volume ratios in the range $[2,5]$, small proteins such as Sec13, Pom34, Nup82-1, Nup82-2, Nup84, Nup85 have worse values.

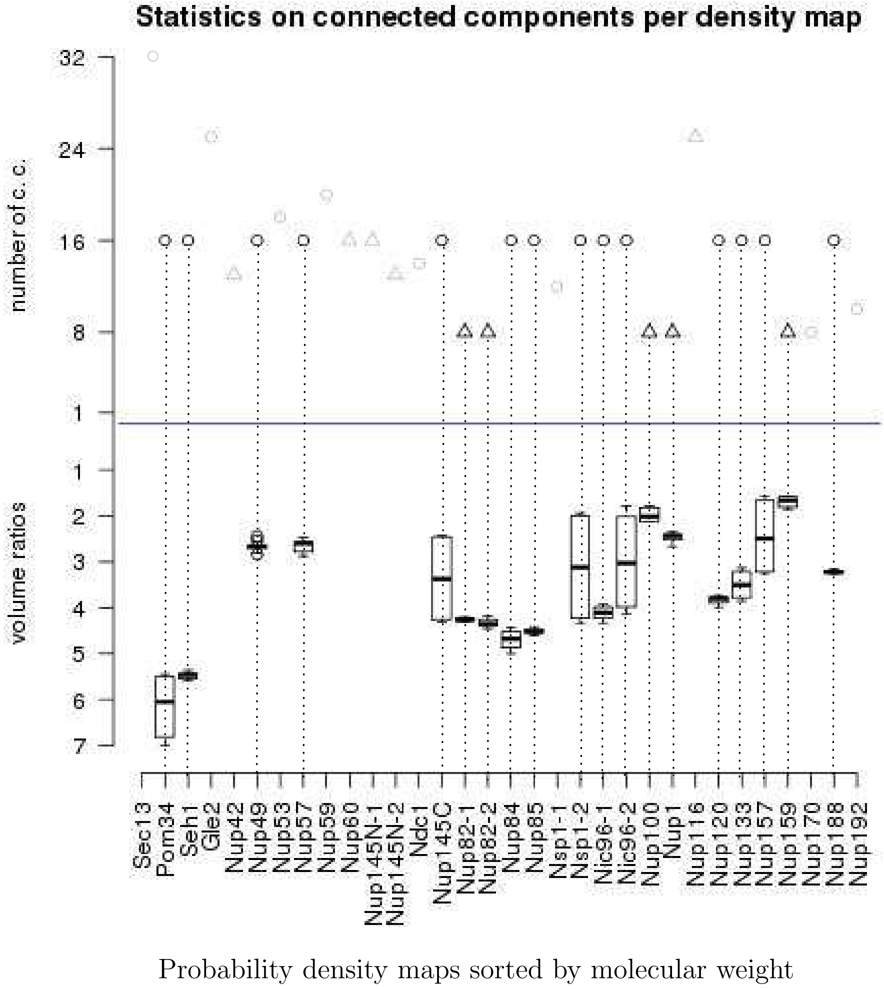


Figure 4.4: Assessing the quality of probability density maps. The names of the 32 maps, with duplicate maps for Nup82, Nsp1, Nic96 and Nup145N, are featured along the abscissa, and are sorted by increasing molecular weight (from 33.0×10^3 for Sec13 to 191.5×10^3 for Nup192, see Table 4.2). **Upper-part.** Number of connected components of voxels with non null density per density map—except for Pom152 which has a single c.c.. Disks correspond to maps with a stoichiometry of 16, while triangles correspond to a stoichiometry of 8. The 19 maps with black marks exhibit the expected stoichiometry, as opposed to the 13 maps with grey marks. **Lower-part.** Box plots of the volume ratios v_{cc_i} of Eq. (4.1), for density maps with a number of c.c. matching the stoichiometry of the protein type.

4.4 Constructing Toleranced Models

The NPC model of [ADV$^+$07a] involves 30 types, whence 34 maps due to four duplicated types (Nup82, Nsp1, Nic96, Nup145N). The map of Gle1 being missing from `http://salilab.org/npc/`, we use the remaining 33 maps as input, for a total of 29 types. We build a toleranced model for each type and merge them to obtain a toleranced model of the whole NPC. Our toleranced model of the NPC comes from the superposition of the toleranced models of the individual protein species. A probability density map involving n protein instances is processed in two stages: first, the map is segmented into n connected regions of voxels, one per protein instance; second, each such region is approximated by a canonical configuration of toleranced balls. In the sequel, we explain the two-stage process.

4.4.1 Allocating Occupancy Volumes

To begin with, we collect voxels in such a way that the volume covered by these voxels matches the estimated volume of all instances, namely Vol_{ref} multiplied by the stoichiometry. For each instance, the set of voxels is called the *occupancy volume*. These voxels are collected by a greedy region growing strategy [ADV$^+$07a, Caption of Figure 9, page 691]. More precisely, this strategy consists of incrementally enlarging a growing region V by adding to it the neighboring voxels of V maximizing the density. Note that the number of collected voxels depends on their volume, which is $1nm^3$ for the input maps. Note also that the greedy algorithm requires a starting point for each protein instance p_i, called its *seed*. We now explain how to select such seeds.
To describe the seed selection, we use the following terminology. The neighbors of a voxel in the map are the 26 adjacent neighbors—by a face, an edge or a vertex. Consider a connected component A of voxels with identical density d. Any voxel of A is called a *local maximum* iff any voxel in the neighborhood of A has a density strictly less than d. We call a *candidate region* a connected component of local maxima. Note that these definitions aims at dealing with extended local maxima rather than local maxima reducing to an isolated voxel.

We apply a greedy strategy which consists of iteratively selecting the candidate regions with top priority, and of updating the priorities of the remaining candidates. To define the priority and the update, denote S_i the set of regions selected after i steps. To define the priority of a candidate region r at step i, consider the two ingredients:

- the density $dens(r)$ of the region;
- the minimum Hausdorff distance [3] from r to the regions already selected, namely $H_i(r) = \min_{s \in S_{i-1}} H(r,s)$

Given two candidate regions r and s, we say that s is dominated by r iff $dens(s) < dens(r)$ and $H_i(r) < H_i(s)$. With these ingredients, the priority of a candidate region r is the number of candidate regions which are dominated by r. The algorithm consists of iteratively selecting candidate region with top priority, and the update step consists of updating the priorities of the remaining candidates.

This strategy calls for two comments:

- Maximizing the Hausdorff distance from a candidate to the regions already selected aims at privileging the repartition and the symmetry of protein instances of the same type.
- The choice of the initial seed may or may not be problematic. For a map featuring a number of candidate regions identical to the stoichiometry of the protein, the output is unique in any case. If the number of candidate regions is larger than the stoichiometry, the placement of protein instances does not admit a unique solution anyway.

4.4.2 Using Canonical Shapes

Creating Instances

Having allocating occupancy volumes, we compute a canonical representation involving 18 toleranced balls for each instance. (In [ADV$^+$07a], at most 13 balls are used to represent a protein instance.) To see how a canonical shape is assigned, consider an occupancy volume O_V to be covered with 18 toleranced balls of identical radius. We perform a principal component analysis (PCA) of the centers of the voxels in O_V, from which we derive three couples (eigen value, eigen vector): $(v_1(O_V), e_1(O_V))$, $(v_2(O_V), e_2(O_V))$ and $(v_3(O_V), e_3(O_V))$. Consider now the three couples obtained from the PCA of the centers of a canonical configuration, denoted $(v_1(C_S), e_1(C_S))$, $(v_2(C_S), e_2(C_S))$ and $(v_3(C_S), e_3(C_S))$. Let σ be a permutation of the symmetric group of size 3—there are 6 such permutations. We determine which canonical shape best represents the volume O_V by selecting the permutation minimizing the following sum:

$$\Sigma_{i=1}^{i=3}(v_{\sigma(i)}(C_S)e_{\sigma(i)}(C_S) - v_i(O_V)e_i(O_V))^2 \tag{4.2}$$

An example of the three over four canonical shapes represented in the TOM of the NPC is given in Figure 4.5.

[3] The Hausdorff distance between two sets A and B is: $\max\{\sup_{a \in A} \inf_{b \in B} d(a,b), \sup_{b \in B} \inf_{a \in A} d(b,a)\}$

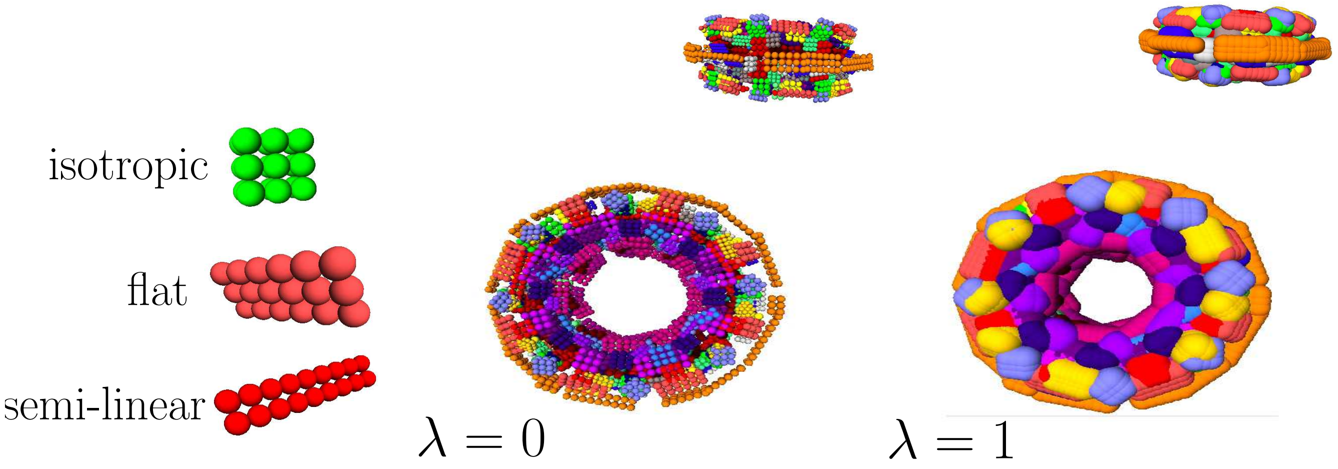


Figure 4.5: Toleranced model of the whole NPC. **(Left.)** The three canonical configurations represented in the toleranced model, 18 balls each, illustrated with protein types Nup120 (isotropic), Nup133 (flat) and Nup84 (semi-linear). **(Middle / Right.)** Views of the inner balls (middle, $\lambda = 0$), and outer balls (right, $\lambda = 1$).

Setting Inner and Outer Radii

For a given protein type, the inner radius is set so that the volume of the union of the 18 inner balls matches the estimated volume of the protein Vol_{ref}. Since the probability density maps of large proteins tend to be more accurate than those of small proteins, see Figure 4.4, we set the outer radius such that the discrepancy $r_i^+ - r_i^-$ is proportional to α / r_i^-:

$$r_i^+ = \frac{\alpha}{r_i^-} + r_i^- . \tag{4.3}$$

Equation (4.3) provides a parametrization of the outer radius as a function of α and r_i^-. Consider a collection of toleranced balls whose outer radii are set this way, that is $\{\overline{B_i}(c_i; r_i^-; r_i^+ = \frac{\alpha}{r_i^-} + r_i^-)\}$. Under the assumption $r_i^+ = \alpha / r_i^- + r_i^-$, the equation (3.3) becomes

$$\lambda(\overline{B_i}, x) = \frac{r_i^-}{\alpha}(\| \; c_i - x \; \| - r_i^-). \tag{4.4}$$

If one equates two such equations to define a Voronoi bisector, that is $\lambda(\overline{B_i}, x) = \lambda(\overline{B_j}, x)$, the α cancel out. Phrased differently, the CW VD of the toleranced balls does not depend on α. Therefore, we arbitrarily set $\alpha = 10$ and compute the whole λ-complex of the toleranced model.

Stopping the Growth Process

Stopping the growth process is naturally a critical issue. Given that our incentive is to account for the uncertainties contained in the input data, the probability density maps for us, a natural strategy consists of stopping the growth when the volume occupied by the grown proteins is too large. To this end, we define the *volume ratio* of a protein in the toleranced model as the volume occupied by the restrictions of the balls defining this protein in the compoundly weighted Voronoi diagram of the toleranced model, divided by the reference volume $Vol_{ref}(P)$ of that protein. (Practically, as computing the volume of restrictions in the compoundly weighted Voronoi diagram is an open problem, we compute the volume of restrictions in the power diagram, see [CKL11].) To examine models with decent geometric accuracy, we stop the growth process at $\lambda = \lambda_{\max}$ when the smallest volume ratio of all instances is larger than the largest volume ratio observed for density maps in Section 4.3.2. Practically, we wet $\lambda_{\max} = 1$, which corresponds to a volume ratio of $\overline{V}_{\lambda_{\max}} \geq 7$.

4.4.3 Assessing Toleranced Models

Definitions

We now wish to assess the geometric accuracy of the toleranced model of section 4.4.2, by comparing sub-complexes with known crystal structures. To this end, given a crystal structure, term a sub-complex encountered along the growth process of *compliant* provided that it contains protein instances of the types found in the crystal structure. As seen from Table 4.1, we compare:

- Vol_{ref} The reference volume.

- $V_{r=0}, V_{r=1.4}$ Consider the Van der Waals model of a known crystal structure. We compute the volume $V_{r=0}$ of this model, and the volume $V_{r=1.4}$ of the associated Solvent Accessible model, namely the model obtained by expanding the VdW radii of 1.4Å. These volumes are meant to provide references for the volumes of toleranced models.

- $V_{c,\lambda_{first}}, V_{c,\lambda_{last}}$ The volume of two compliant sub-complexes, namely the first one and the last one encountered along the growth process, respectively encountered at $\lambda = \lambda_{first}$ and $\lambda = \lambda_{last}$. Note that these complexes are spotted from the Hasse diagram. Following the volume ratio of Eq. (5.1), the volume of a compliant sub-complex is computed as the sum of the volumes of its Voronoi restrictions in the power diagram [CKL11], using our software Vorlume, see `http://cgal.inria.fr/abs/Vorlume/`. These volumes, expressed in nm^3, are denoted $V_{c,\lambda_{first}}$ and $V_{c,\lambda_{last}}$.

Crystal Structures versus Reference Volumes

The upper left region of Table 4.1 compares the reference volume to $V_{r=0}$ and $V_{r=1.4}$. Regarding Van der Waals models, the ratio $V_{r=0}/Vol_{ref}$ lies in the range 0.33 - 0.49, respectively for the Y_X-edge and Nic96, showing that Van der Walls volumes underestimate the volume of globular proteins. On the other hand, except for Nup133 and the Y_X-tail, the ratio $V_{r=1.4}/Vol_{ref}$ lies in the range 0.65 - 1.02, values respectively attained for the Y_X-edge and Nic96. Thus, Solvent Accessible models on a per-atom basis provide a relatively good approximation of reference volumes estimated on a per-residue basis.

Reference Volumes versus Volumes of Compliant Sub-Complexes

As seen from the upper-right region of Table 4.1, except for three copies of Y_X-long-arm, all compliant sub-complexes appear at $\lambda = 0$ with a volume ratio varying in the range 0.77 - 0.97 for the Y_X-long-arm and the Y_X-short-arm. We note that these values are comparable to those of Solvent Accessible models. (As explained in section 4.4, the inner radius is set such that the volume ratio of a compliant sub-complex of a protein for $\lambda = 0$ is equal to one. The values observed, which are less than one, are due to overlaps with other protein instances.)

The lower-right region of Table 4.1 reports these ratios for sub-complexes of Y-complex and T-complex with no known crystal structure. All compliant sub-complexes of the Y-complex appear with a volume ratio in the range 0.83 - 2.22 for the Y-arms and the Y-main. For the T-complex, though, compliant sub-complexes have a volume ratio in the range 0.17 - 1.49 for the T-leg and the T-core. The lower bound for the T-leg corresponds to a copy partially covered by the remaining toleranced proteins of the NPC, whence a considerably reduced volume.

Protein types	ref.	PDB id	Res ($\mathring{A}$)	$\frac{V_{r=0}}{Vol_{ref}}$	$\frac{V_{r=1.4}}{Vol_{ref}}$	Vol_{ref}	λ_{first}	$\frac{V_{c,\lambda_{first}}}{Vol_{ref}}$	λ_{last}	$\frac{V_{c,\lambda_{last}}}{Vol_{ref}}$
Y_X-edge	[NHD+09]	3IKO	3.20	0.33	0.66	324.3	0	0.83	0	0.87
	[BS09]	3JRO	4.00	0.31	0.65	324.3	0	0.83	0	0.87
Y_X-long-arm	[DMS+08]	3F3F	2.90	0.48	0.97	153.3	0	0.77	0.17	1.65
	[BLS+08]	3EWE	3.50	0.37	0.79	153.3	0	0.77	0.17	1.65
Y_X-short-arm	[SMD+09]	3F7F	2.60	0.45	0.91	149.8	0	0.89	0	0.97
	[SMD+09]	3H7N	3.00	0.45	0.91	149.8	0	0.89	0	0.97
	[LBS09]	3HXR	3.00	0.41	0.85	149.8	0	0.89	0	0.97
Y_X-tail (homologous)	[WS09]	3I4R	3.53	0.23	0.51	269.9	0	0.79	0	0.89
Nup133 (N-terminal)	[SMD+04]	1XKS	2.35	0.20	0.42	165.7	0	0.82	0	0.91
Nic96	[JS07]	2QX5	2.50	0.45	0.92	119.9	0	0.77	0	0.88
Nic96	[SSF+08]	2RFO	2.60	0.49	1.02	119.9	0	0.77	0	0.88
Y-arms						302.1	0	0.83	0	0.93
Y-junction						434.6	0.04	1.14	0.58	2.07
Y-core						538.8	0	0.86	0.44	2.18
Y-main						704.5	0	0.86	0.44	2.22
Y-complex						793.1	0.04	1.11	0.21	1.85
T-leg						354.8	0	0.17	0	0.27
T-core						224.2	0	0.79	0.15	1.49
T-complex						579.0	0	0.48	0.15	0.78

Table 4.1: Comparison of volumes of selected proteins and sub-complexes : crystal structures versus toleranced models. **Top.** Crystal structures versus toleranced models of sub-complexes of the Y-complex and the T-complex. **Bottom.** Toleranced models of interesting sub-complexes of the NPC.

On the canonical shapes of protein instances

In [ADV+07a], a protein model consists of tangent balls, and up to nine representations of varying resolution (i.e. number of balls) are used per protein type, at different stages of the reconstruction algorithm—see Fig. 1.5 in Chapter 1. The number of balls for the finest resolution is determined by the nearest integer value of the axial ratio of a prolate ellipsoid (a rugby ball) computed from the sedimentation coefficient of the protein. The details can be found in [ADV+07a, Supplemental table 5, page 92], and the maximum number of balls is reproduced in the last column of Table 4.2. We note in passing that n tangent balls positioned along a line-segment yield a maximum elongation ratio of $n:1$. The NPC reconstruction algorithm distorts i.e. folds these initials representations; unfortunately, Sali et al. [ADV+07a] do not report any information of the final configurations of balls.

Comparing the finest representation of Alber et al against our four canonical configurations in the toleranced model calls for the following comments. First, the linear configuration case is not found in the toleranced model. Second, a number of mismatches are possibly observed, in particular when Alber et al use elongated models while our instances are flat or roughly isotropic. (Here, *possibly* just means that the assessment is based on the initial shapes of the proteins in Alber et al, not on the final ones—which we do not know.) To understand these potential discrepancies, we examine three cases—see Table 4.2 for the number of instances of each type in the toleranced model:

- Nup116, see Figure 4.26. Alber et al use 13 balls while we observe a repartition of (0,0,1,7) for the four canonical shapes. This mismatch clearly comes from the quality of the density map. First, it is very disconnected and the 25 connected components of the map do not account for a stoichiometry of eight. Second, the eight sets of voxels selected to represent the eight instances are roughly isotropic.

- Nup159, see Figure 4.31. Alber et al use 11 balls while we observe a repartition of (0,0,4,4). Visual inspection of the map shows that connected components are well resolved; yet the geometry of the components do not support an elongation ratio of 11:1.

- Nup100, see in Figure 4.24. Alber et al use 13 balls while we observe a repartition of (0,0,0,8). This case is similarly to that of Nup159, as the well resolved-ness of the map contrasts with the 11:1 ratio.

Protein type	Average Mol. Weight ($\times 10^3$)	Stoich.	#linear	#semi linear	#flat	#roughly isotropic	#balls in [ADV$^+$07b]
Nup192	191.5	16	0	0	0	16	2
Nup188	188.6	16	0	0	0	16	2
Nup170	169.5	16	0	0	7	9	2
Nup159*	158.9	8	0	0	4	4	11
Nup157	156.6	16	0	0	0	16	3
Pom152	151.7	16	0	6	8	2	10
Nup133	133.3	16	0	0	6	10	2
Nup120	120.4	16	0	0	2	14	2
Nup116*	116.2	8	0	0	0	8	13
Nup1*	113.6	8	0	0	4	4	9
Nup100*	100.0	8	0	0	0	8	13
Nic96-1	96.2	16	0	0	0	16	2
Nic96-2	96.2	16	0	0	0	16	2
Nsp1-1*	86.5	16	0	0	10	6	12
Nsp1-2*	86.5	16	0	0	0	16	12
Nup85	84.9	16	0	0	0	16	3
Nup84	83.6	16	0	1	5	10	3
Nup82-1	82.1	8	0	0	1	7	2
Nup82-2	82.1	8	0	0	0	8	2
Nup145C	81.1	16	0	0	0	16	2
Ndc1	74.1	16	0	1	7	8	2
Nup145N-1*	64.6	8	0	0	0	8	6
Nup145N-2*	64.6	8	0	0	0	8	6
Nup60*	59.0	8	0	0	0	8	4
Nup59*	58.8	16	0	0	6	10	4
Nup57*	57.5	16	0	0	0	16	3
Nup53*	52.6	16	0	0	6	10	3
Nup49*	49.1	16	0	0	0	16	3
Nup42*	42.8	8	0	0	3	5	5
Gle2	40.5	16	0	0	1	15	1
Seh1	39.1	16	0	0	0	16	1
Pom34	34.2	16	0	1	13	2	3
Sec13	33.0	16	0	0	4	12	1
Total	NA	448	0	9	86	353	NA

Table 4.2: Protein types sorted by decreasing average molecular weights (no dimension, first column), their stoichiometry (2nd column), the number of instances for the four canonical shapes in the toleranced model of the NPC (columns 3-6), and the number of beads used at the finest representation level by Alber et al

4.5 Supplemental

4.5.1 Probability Density Map Pictures

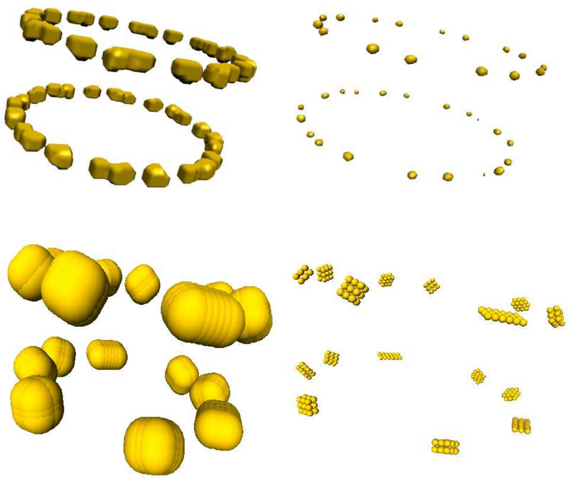


Figure 4.6: Sec13 ($Vol_{ref} = 40.7nm^3$) **Top.** Probability density map with all no null voxels (**left**) and with all voxels with a density larger than the half of the maximum density (**right**). There are 32 c.c. **Bottom.** Toleranced proteins of Sec13 with their outer balls (**left**) and their inner balls (**right**).

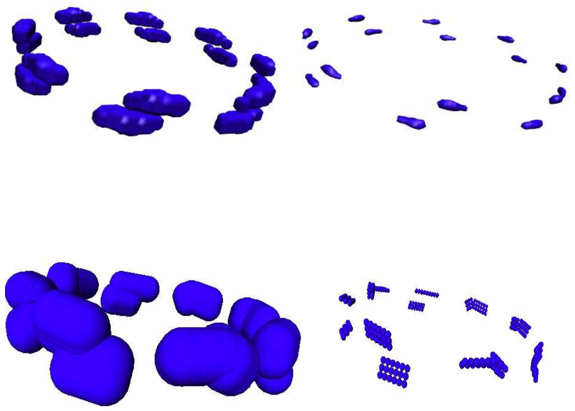


Figure 4.7: Pom34 ($Vol_{ref} = 42.627nm^3$) **Top.** Probability density map with all no null voxels (**left**) and with all voxels with a density larger than the half of the maximum density (**right**). There are 32 c.c. **Bottom.** Toleranced proteins of Pom34 with their outer balls (**left**) and their inner balls (**right**).

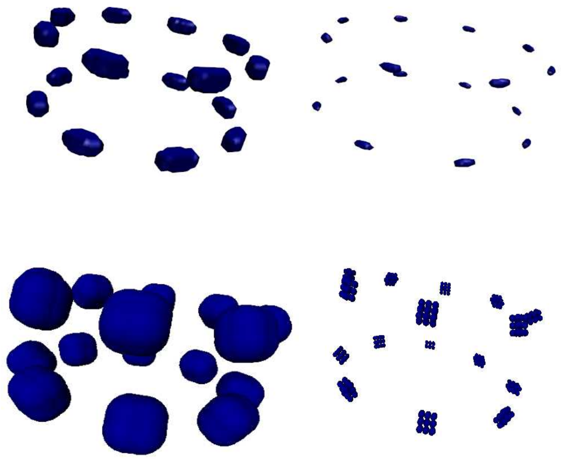


Figure 4.8: Seh1 ($Vol_{ref} = 47.892nm^3$) **Top.** Probability density map with all no null voxels (**left**) and with all voxels with a density larger than the half of the maximum density (**right**). There are 16 c.c. **Bottom.** Toleranced proteins of Seh1 with their outer balls (**left**) and their inner balls (**right**).

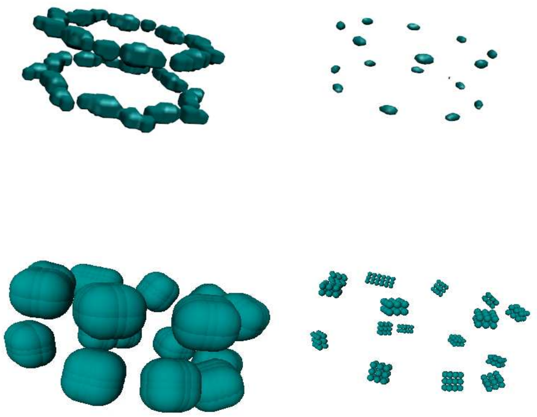


Figure 4.9: Gle2 ($Vol_{ref} = 49.6nm^3$) **Top.** Probability density map with all no null voxels (**left**) and with all voxels with a density larger than the half of the maximum density (**right**). There are 25 c.c. **Bottom.** Toleranced proteins of Gle2 with their outer balls (**left**) and their inner balls (**right**).

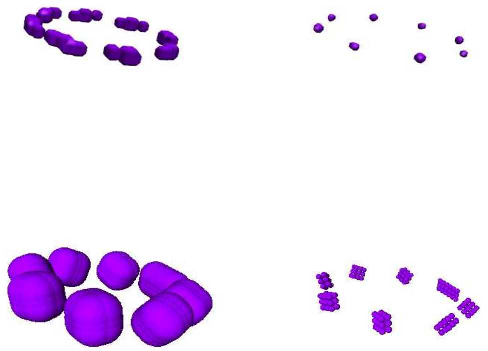


Figure 4.10: Nup42 ($Vol_{ref} = 51.853nm^3$) **Top.** Probability density map with all no null voxels (**left**) and with all voxels with a density larger than the half of the maximum density (**right**). There are 13 c.c. **Bottom.** Toleranced proteins of Nup42 with their outer balls (**left**) and their inner balls (**right**).

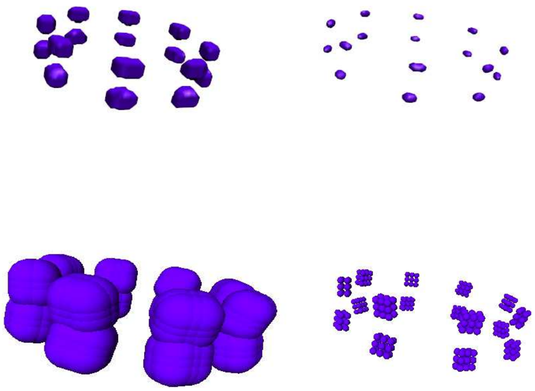


Figure 4.11: Nup49 ($Vol_{ref} = 60.199nm^3$) **Top.** Probability density map with all no null voxels (**left**) and with all voxels with a density larger than the half of the maximum density (**right**). There are 16 c.c. **Bottom.** Toleranced proteins of Nup49 with their outer balls (**left**) and their inner balls (**right**).

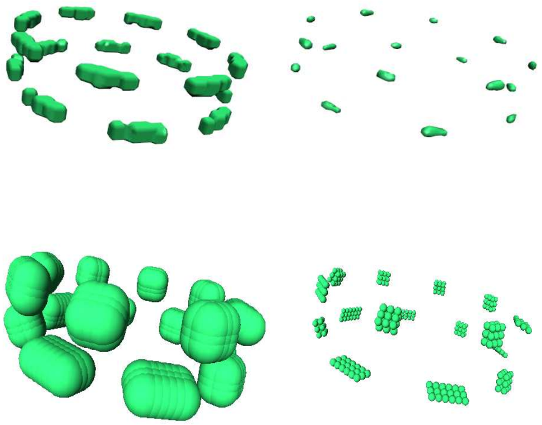


Figure 4.12: Nup53 ($Vol_{ref} = 64.695nm^3$) **Top.** Probability density map with all no null voxels (**left**) and with all voxels with a density larger than the half of the maximum density (**right**). There are 18 c.c. **Bottom.** Toleranced proteins of Nup53 with their outer balls (**left**) and their inner balls (**right**).

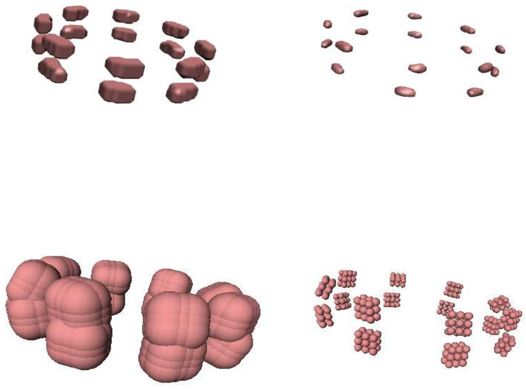


Figure 4.13: Nup57 ($Vol_{ref} = 70.401nm^3$) **Top.** Probability density map with all no null voxels (**left**) and with all voxels with a density larger than the half of the maximum density (**right**). There are 16 c.c. **Bottom.** Toleranced proteins of Nup57 with their outer balls (**left**) and their inner balls (**right**).

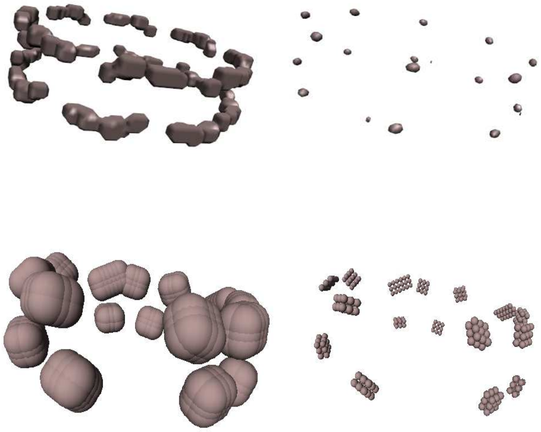


Figure 4.14: Nup59 ($Vol_{ref} = 71.3nm^3$) **Top.** Probability density map with all no null voxels (**left**) and with all voxels with a density larger than the half of the maximum density (**right**). There are 20 c.c. **Bottom.** Toleranced proteins of Nup59 with their outer balls (**left**) and their inner balls (**right**).

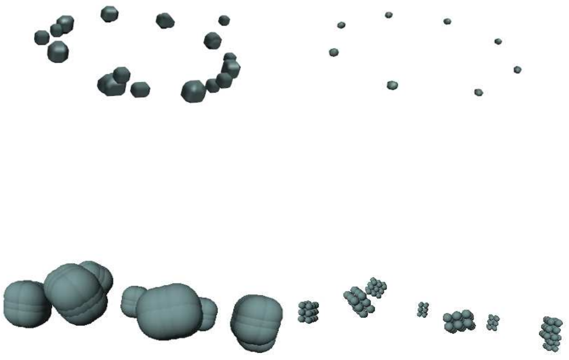


Figure 4.15: Nup60 ($Vol_{ref} = 72.133nm^3$) **Top.** Probability density map with all no null voxels (**left**) and with all voxels with a density larger than the half of the maximum density (**right**). There are 16 c.c. **Bottom.** Toleranced proteins of Nup60 with their outer balls (**left**) and their inner balls (**right**).

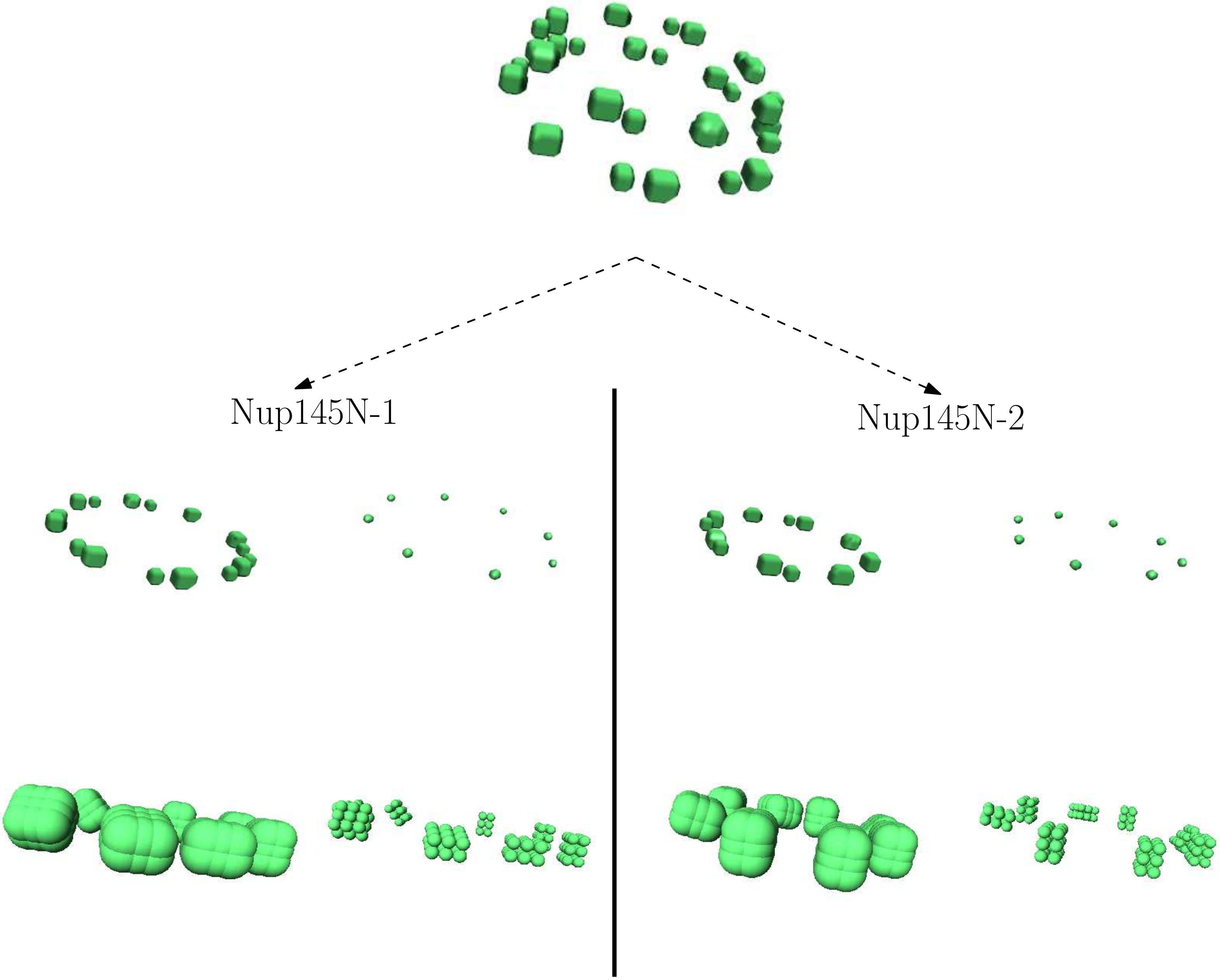


Figure 4.16: Nup145N ($Vol_{ref} = 179.4nm^3$) **Top.** Combined probability density maps of Nup145N-1 and Nup145N-2 with all no null voxels. **Nup145N-1** : probability density map with all no null voxels (**top left**), and with voxels having density larger than the half of maximal density (**top right**), and toleranced proteins with outer balls (**bottom left**) and inner balls (**bottom right**). There are 16 c.c. **Nup145N-2**: probability density map with all no null voxels (**top left**), and with voxels having density larger than the half of maximal density (**top right**), and toleranced proteins with outer balls (**bottom left**) and inner balls (**bottom right)** There are 13 c.c.

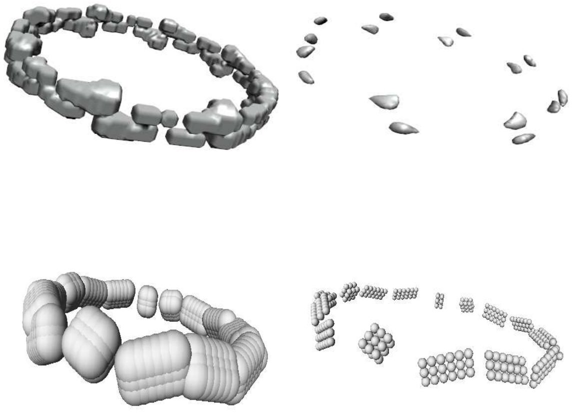


Figure 4.17: Ndc1 ($Vol_{ref} = 92.760nm^3$) **Top.** Probability density map with all no null voxels (**left**) and with all voxels with a density larger than the half of the maximum density (**right**). There are 14 c.c. **Bottom.** Toleranced proteins of Ndc1 with their outer balls (**left**) and their inner balls (**right**).

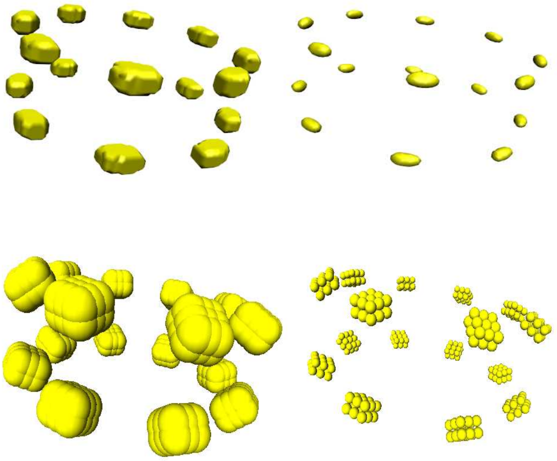


Figure 4.18: Nup145C ($Vol_{ref} = 179.373nm^3$) **Top.** Probability density map with all no null voxels (**left**) and with all voxels with a density larger than the half of the maximum density (**right**). There are 16 c.c. **Bottom.** Toleranced proteins of Nup145C with their outer balls (**left**) and their inner balls (**right**).

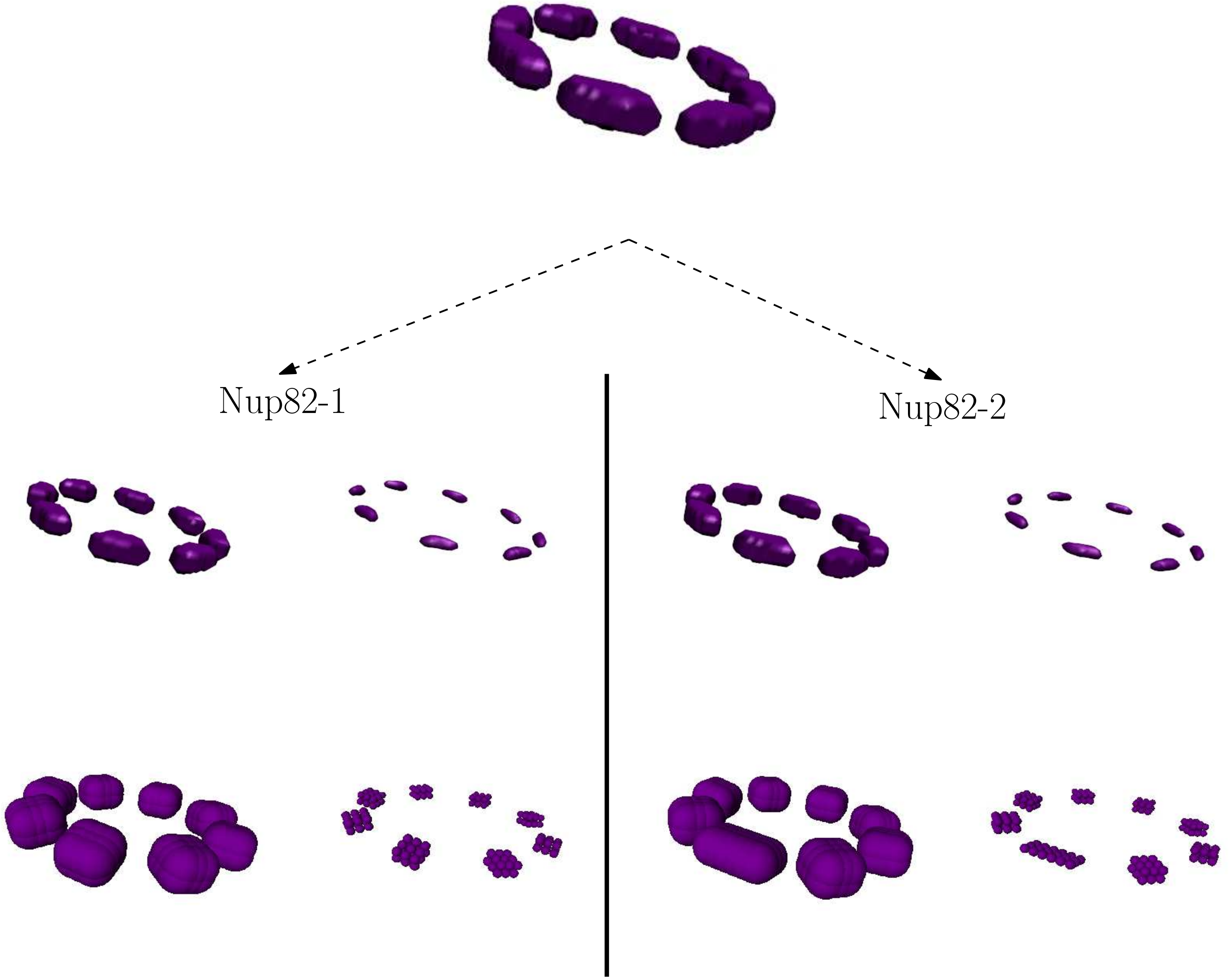


Figure 4.19: Nup82 ($Vol_{ref} = 101.9nm^3$) **Top.** Combined probability density maps of Nup82-1 and Nup82-2 with all no null voxels. **Nup82-1**: probability density map with all no null voxels (**top left**), and with voxels having density larger than the half of maximal density (**top right**), and toleranced proteins with outer balls (**bottom left**) and inner balls (**bottom right**). There are 8 c.c. **Nup82-2**: probability density map with all no null voxels (**top left**), and with voxels having density larger than the half of maximal density (**top right**), and toleranced proteins with outer balls (**bottom left**) and inner balls (**bottom right)** There are 8 c.c.

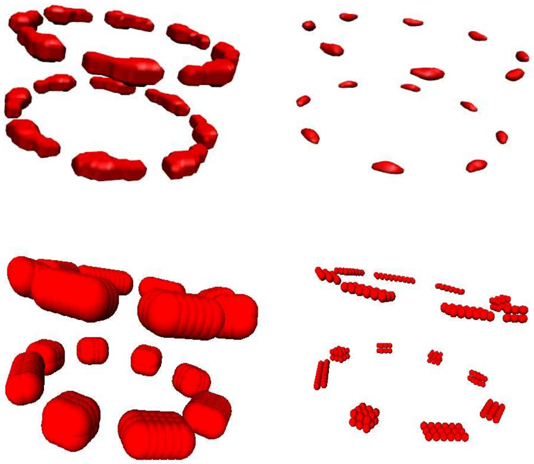


Figure 4.20: Nup84 ($Vol_{ref} = 104.171nm^3$) **Top.** Probability density map with all no null voxels (**left**) and with all voxels with a density larger than the half of the maximum density (**right**). There are 16 c.c. **Bottom.** Toleranced proteins of Nup84 with their outer balls (**left**) and their inner balls (**right**).

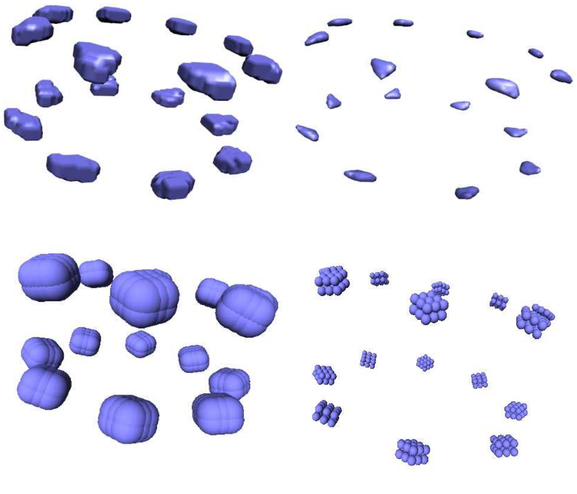


Figure 4.21: Nup85 ($Vol_{ref} = 105.416nm^3$) **Top.** Probability density map with all no null voxels (**left**) and with all voxels with a density larger than the half of the maximum density (**right**). There are 16 c.c. **Bottom.** Toleranced proteins of Nup85 with their outer balls (**left**) and their inner balls (**right**).

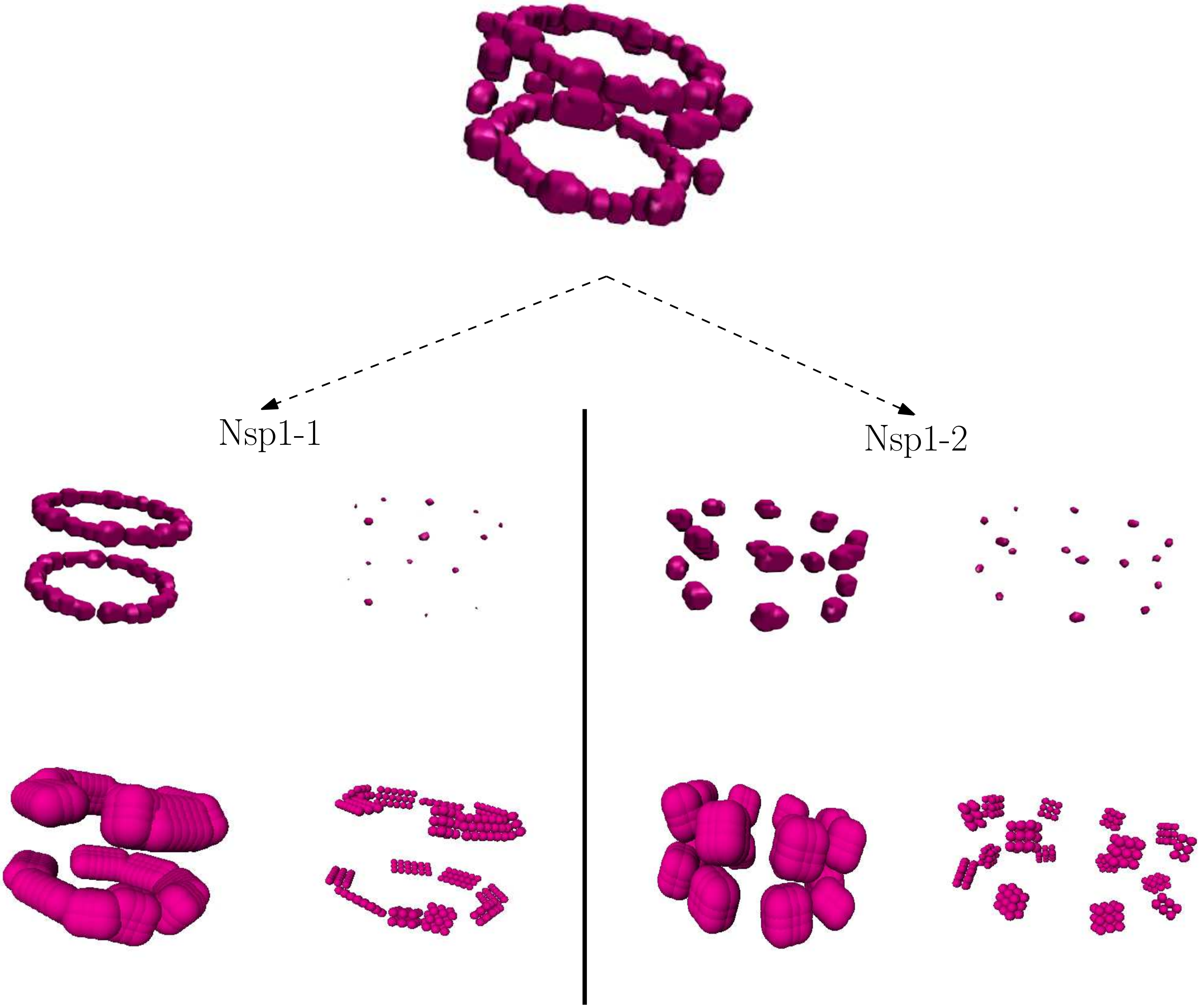


Figure 4.22: Nsp1 ($Vol_{ref} = 104.4nm^3$) **Top.** Combined probability density maps of Nsp1-1 and Nsp1-2 with all no null voxels. **Nsp1-1**: probability density map with all no null voxels (**top left**), and with voxels having density larger than the half of maximal density(**top right**), and toleranced proteins with outer balls (**bottom left**) and inner balls (**bottom right**). There are 12 c.c. **Nsp1-2**: probability density map with all no null voxels (**top left**), and with voxels having density larger than the half of maximal density (**top right**), and toleranced proteins with outer balls (**bottom left**) and inner balls (**bottom right)** There are 16 c.c.

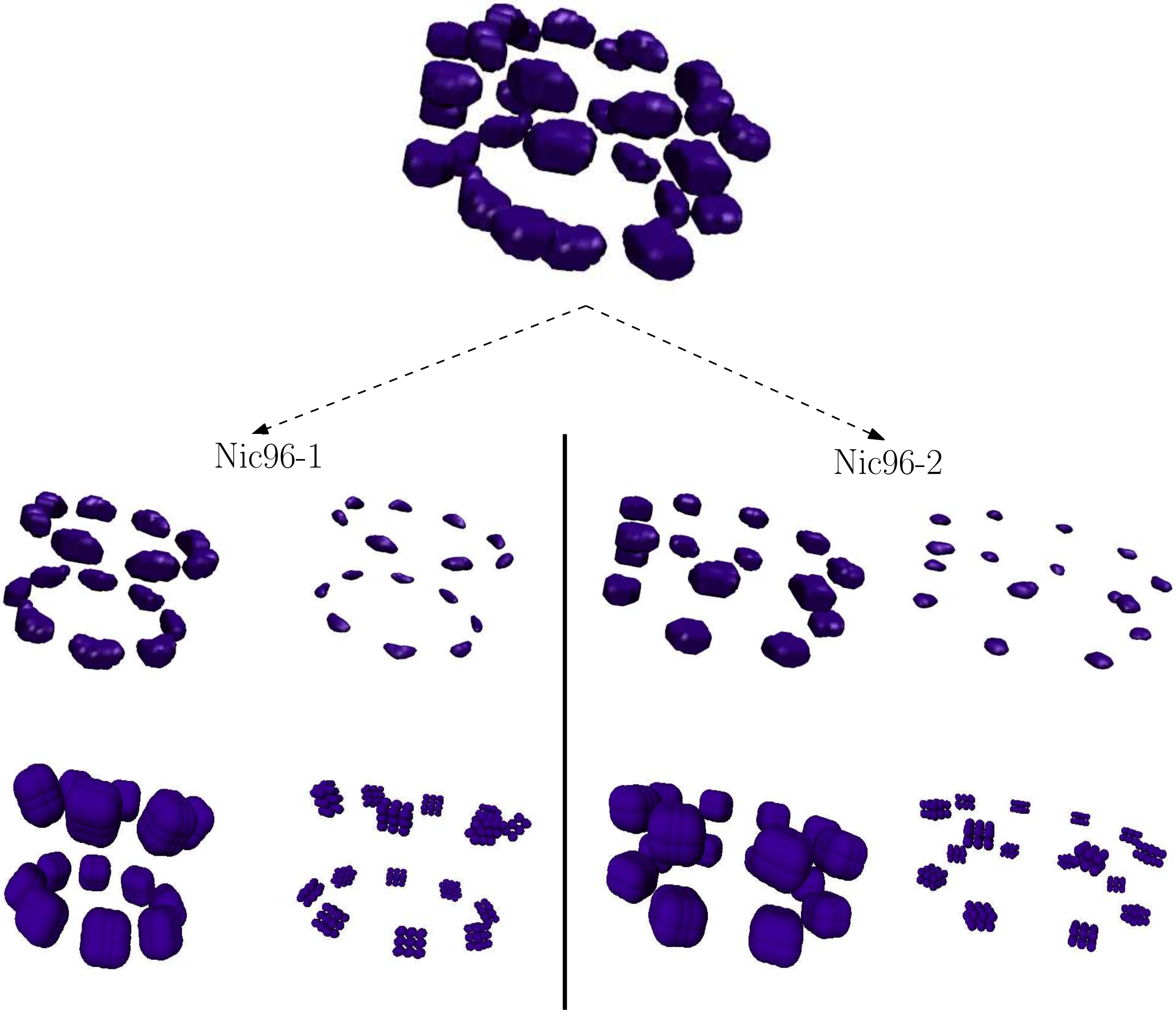


Figure 4.23: Nic96 ($Vol_{ref} = 119.9nm^3$) **Top.** Combined probability density maps of Nic96-1 and Nic96-2 with all no null voxels. **Nic96-1**: probability density map with all no null voxels (**top left**), and with voxels having density larger than the half of maximal density (**top right**), and toleranced proteins with outer balls (**bottom left**) and inner balls (**bottom right**). There are 16 c.c. **Nic96-2**: probability density map with all no null voxels (**top left**), and with voxels having density larger than the half of maximal density (**top right**), and toleranced proteins with outer balls (**bottom left**) and inner balls (**bottom right**) There are 16 c.c.

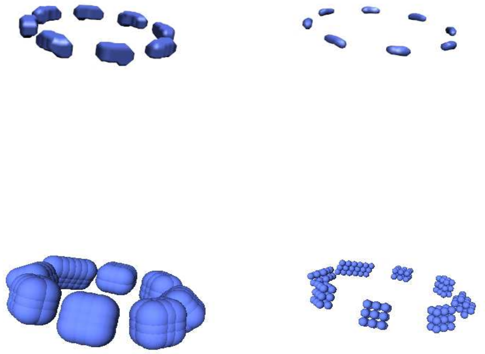


Figure 4.24: Nup100 ($Vol_{ref} = 121.039nm^3$) **Top.** Probability density map with all no null voxels (**left**) and with all voxels with a density larger than the half of the maximum density (**right**). There are 8 c.c. **Bottom.** Toleranced proteins of Nup100 with their outer balls (**left**) and their inner balls (**right**).

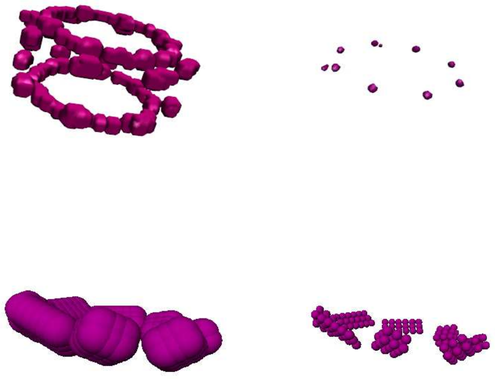


Figure 4.25: Nup1 ($Vol_{ref} = 138.103nm^3$) **Top.** Probability density map with all no null voxels (**left**) and with all voxels with a density larger than the half of the maximum density (**right**). There are 8 c.c. **Bottom.** Toleranced proteins of Nup1 with their outer balls (**left**) and their inner balls (**right**).

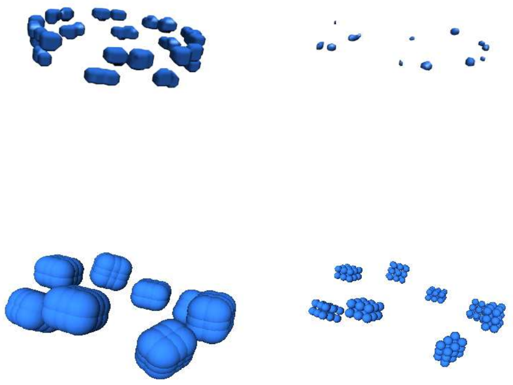


Figure 4.26: Nup116 ($Vol_{ref} = 141.053nm^3$) **Top.** Probability density map with all no null voxels (**left**) and with all voxels with a density larger than the half of the maximum density (**right**). There are 25 c.c. **Bottom.** Toleranced proteins of Nup116 with their outer balls (**left**) and their inner balls (**right**).

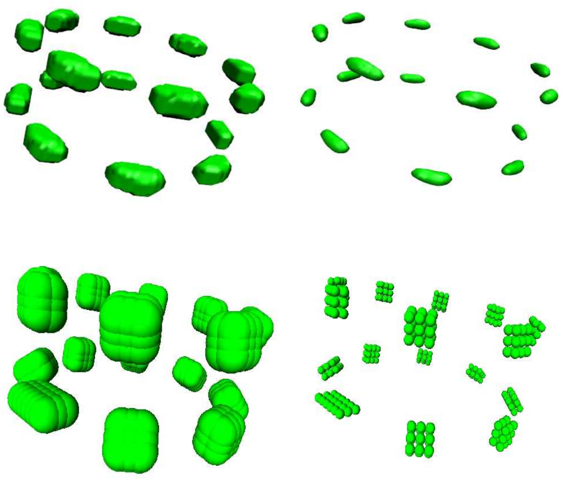


Figure 4.27: Nup120 ($Vol_{ref} = 149.8nm^3$) **Top.** Probability density map with all no null voxels (**left**) and with all voxels with a density larger than the half of the maximum density (**right**). There are 16 c.c. **Bottom.** Toleranced proteins of Nup120 with their outer balls (**left**) and their inner balls (**right**).

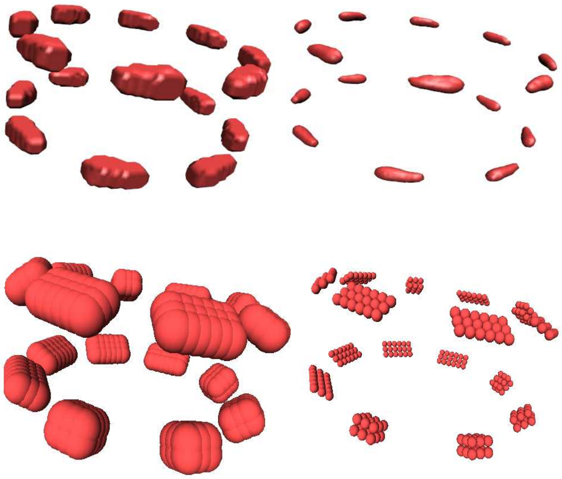


Figure 4.28: Nup133 ($Vol_{ref} = 165.734nm^3$) **Top.** Probability density map with all no null voxels (**left**) and with all voxels with a density larger than the half of the maximum density (**right**). There are 16 c.c. **Bottom.** Toleranced proteins of Nup133 with their outer balls (**left**) and their inner balls (**right**).

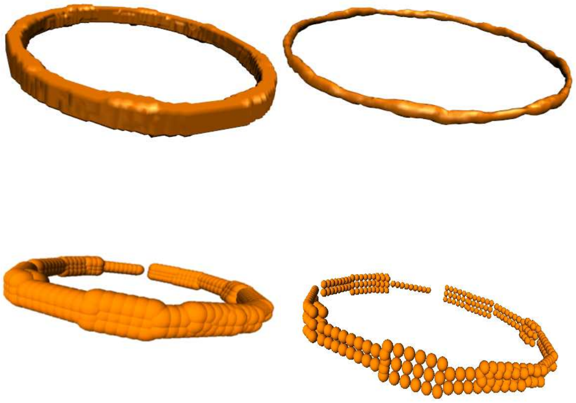


Figure 4.29: Pom152 ($Vol_{ref} = 188.354nm^3$) **Top.** Probability density map with all no null voxels (**left**) and with all voxels with a density larger than the half of the maximum density (**right**). There is 1 c.c. **Bottom.** Toleranced proteins of Pom152 with their outer balls (**left**) and their inner balls (**right**).

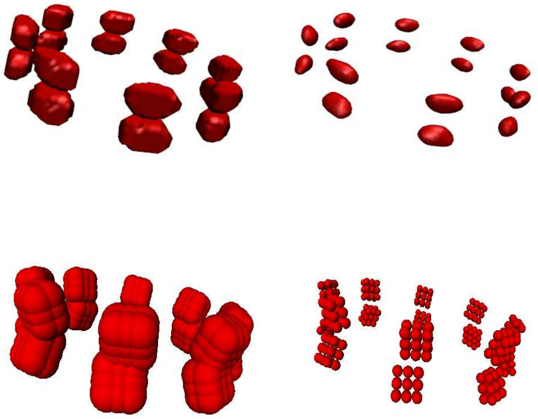


Figure 4.30: Nup157 ($Vol_{ref} = 194.7nm^3$) **Top.** Probability density map with all no null voxels (**left**) and with all voxels with a density larger than the half of the maximum density (**right**). There are 8 c.c. **Bottom.** Toleranced proteins of Nup157 with their outer balls (**left**) and their inner balls (**right**).

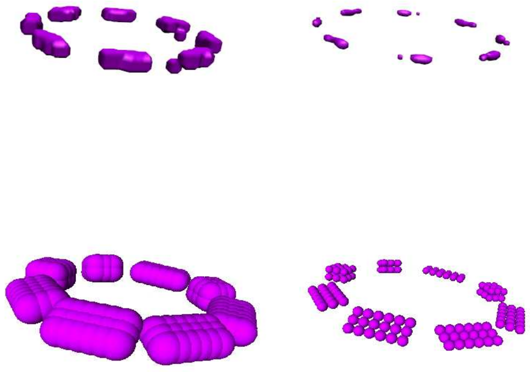


Figure 4.31: Nup159 ($Vol_{ref} = 193.902nm^3$) **Top.** Probability density map with all no null voxels (**left**) and with all voxels with a density larger than the half of the maximum density (**right**). There are 8 c.c. **Bottom.** Toleranced proteins of Nup159 with their outer balls (**left**) and their inner balls (**right**).

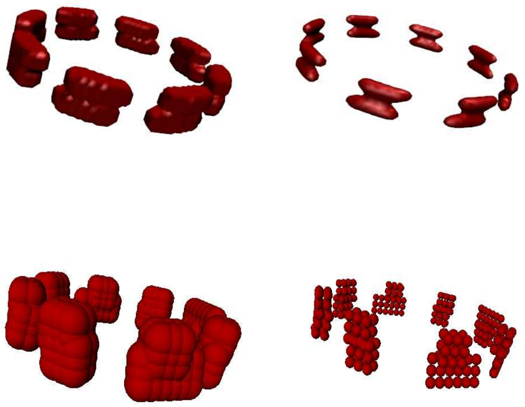


Figure 4.32: Nup170 ($Vol_{ref} = 210.930nm^3$) **Top.** Probability density map with all no null voxels (**left**) and with all voxels with a density larger than the half of the maximum density (**right**). There are 16 c.c. **Bottom.** Toleranced proteins of Nup170 with their outer balls (**left**) and their inner balls (**right**).

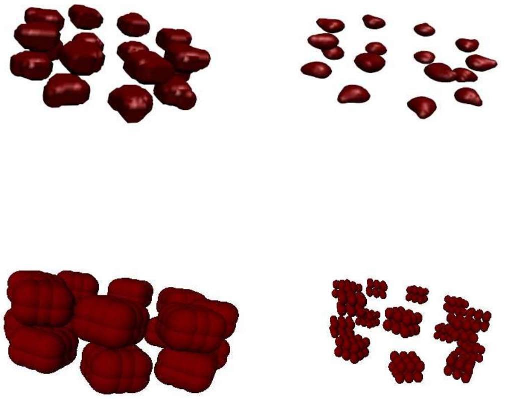


Figure 4.33: Nup188 ($Vol_{ref} = 237.054nm^3$) **Top.** Probability density map with all no null voxels (**left**) and with all voxels with a density larger than the half of the maximum density (**right**). There are 16 c.c. **Bottom.** Toleranced proteins of Nup188 with their outer balls (**left**) and their inner balls (**right**).

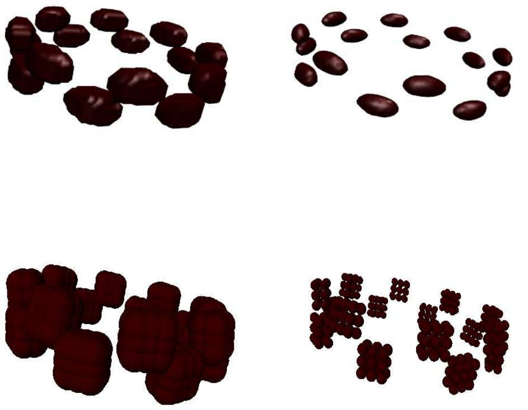


Figure 4.34: Nup192 ($Vol_{ref} = 239.604nm^3$) **Top.** Probability density map with all no null voxels (**left**) and with all voxels with a density larger than the half of the maximum density (**right**). There are 10 c.c. **Bottom.** Toleranced proteins of Nup192 with their outer balls (**left**) and their inner balls (**right**).

Chapter 5

Assessing the Reconstruction of Macro-Molecular Assemblies: Contact Probabilities and Sub-complexes

5.1 Introduction - Rationale

In molecular modeling, facing uncertainties on the shape of proteins and/or on their positions is commonplace. The case of interest in this thesis is that of probability density maps, namely those computed from the reconstructions of the NPC. In section 4.4, we have seen how to construct a toleranced model for such maps, so as to take into account high and low confidence regions. This chapter makes three contributions.
First, we describe in Section 5.2 a panoply of tools to analyze the toleranced model of a macro-molecular assembly. Second, we present in Section 5.3 an analysis of the contacts between protein types, and show that our contact probabilities go beyond the frequencies defined by Sali et al. Third, we develop our analysis over three sub-complexes of the NPC: the Y-complex in Section 5.4, the T-complex in Section 5.5 and the Nup82-complex in Section 5.6.

5.2 Analysis Tools for Toleranced Models of Proteins and Assemblies

5.2.1 Tracking Contacts Between Proteins in Toleranced Models

Toleranced proteins and assemblies. We define a *toleranced protein* (denoted p_j as for protein instances) as a collection of toleranced balls, and a *toleranced assembly* as a collection of toleranced proteins. For a given value of λ, a protein of intermediate size is denoted $\overline{p_j}[\lambda]$, and $\mathscr{F}_\lambda$ denotes the domain corresponding to the union of growing balls, that is $\mathscr{F}_\lambda = \cup_i \overline{B_i}[\lambda] = \cup_j \overline{p_j}[\lambda]$. For a fixed λ, the topology of the domain $\mathscr{F}_\lambda$ is of utmost interest: a connected component of this domain is called a *complex*; the domain is called a *mixture* if it involves several complexes.

This terminology clearly holds when all the protein types are equivalent. Because a number of experiments are conserved with a set of specific protein types, we may also segregate all the protein types into blue and red types. In this *bicolor* setting, we shall focus on connected components involving red proteins only. Again, this setting is meant to deal with models of large assemblies, where the red group will refer to the protein types involved in a TAP experiment or to those seen in a sub-complex. These notions are illustrated on Figure 5.1.

Curved Voronoi diagrams. To compute complexes and mixtures in the bicolor setting, we resort to the theory of curved Voronoi diagrams and α-shapes presented in Chapter 3. Intuitively, the growth process of Eq. (3.3) allows one partition the three-dimensional space of into so-called *Voronoi regions*, with one region V_i for each toleranced ball $\overline{B_i}$: a point x belongs to V_i if the growing ball $\overline{B_i}[\lambda]$ reaches point x before any ball $\overline{B_j}[\lambda] \neq \overline{B_i}[\lambda]$. A region V_i is bounded by curved bisectors defined by $\overline{B_i}$ and neighboring balls.

For a given ball $\overline{B_i}[\lambda]$, consider its *restriction* to its Voronoi region, that is the intersection $\overline{B_i}[\lambda] \cap V_i$. These restrictions naturally partition the domain $\mathscr{F}_\lambda$, and their connected components correspond to the aforementioned complexes. Moreover, we use the pairwise intersections between the restrictions involved in a complex C to define its *skeleton* graph G_C: its nodes are the toleranced proteins of C; an edge links p_i and p_j provided that there exists two intersecting restrictions, one from p_i and one from p_j.

Remark 15. *The pairs of balls reported correspond to balls whose restrictions intersect, and these pairs form a subset of all pairs of intersecting balls. In line with this comment, if ones keeps growing balls all the way to $\lambda = \infty$, one does not end up with all pairs of intersecting balls, but the pairs giving rise to a bisector in the Voronoi diagram defined by the balls. For a large enough value of λ, the pairs obtained are the abstract simplices of the dual complex of the Voronoi diagram.*

λ-complex versus partial λ-complex. In Chapter 3, we have introduced the partial λ-complex, see remark 11. The underlying motivation is of computational nature: the only known algorithm to compute the λ-complex has $O(n^5)$ complexity, while a naive scan of all pairs of toleranced balls yields a computation of Gabriel edges in $O(n^3)$ time—with n the number of toleranced balls.
From a practical standpoint, recall that our model of a half-spoke contains $33 \times 18 = 594$ toleranced balls for 33 protein instances, so that the whole toleranced model of the NPC contains $594 \times 16 = 9504$ toleranced balls. These sizes and complexities explain why the λ-complex can be computed on a half-spoke, but not on the whole NPC, which we processed with the partial λ-complex. However, as explained in supplemental Section 5.8.1, the difference between both complexes is not significant.

5.2.2 Analyzing Proteins and Contacts during the Growth Process

Stability analysis. Growing λ results in merges between complexes. The set of finite topologies [1] corresponding to this evolution can be represented in a directed acyclic graph called *Hasse diagram*, a special graph whose nodes are the complexes, with an edge joining (generically) two nodes when the complexes merge along the growth process. The origin (endpoint) of an edge therefore represents the birth (resp. death) of a complex C: at $\lambda = \lambda_b(C)$, the complex gets formed by a merge of two or more complexes; at $\lambda = \lambda_d(C)$, the complex dies by merging with at least another complex. Thus, the *lifetime* $l(C) = \lambda_d(C) - \lambda_b(C)$ provides a measure of the topological stability of the complex C. Also, the *ancestors* and *successors* of C are the complexes contained into and containing, respectively, the complex C. See Figure 5.1(Bottom right) for an illustration.
In the bicolor setting, let T be the list of red protein types. A complex C of the Hasse diagram is made of instances whose types are in T. If each type of T is present exactly once in C, the complex C is termed an *isolated copy*. The number and the lifetime of isolated copies give a measure of the separability of the different copies of a complex involving all the types of T. Note that the intersection of the lifetime intervals of the different isolated copies may be empty.

Volume ratio. Estimating the volume $Vol_{ref}(p_i)$ of a protein instance p_i from its sequence [HGC94], let $Vol_{ref}(C) = \sum_{p_i \in C} Vol_{ref}(p_i)$ the reference volume of the complex C, estimated from its constituting instances. On the other hand, for a fixed λ, let $Vol_\lambda(C)$ be the volume of the complex C, defined as the sum of the volumes of the Voronoi restrictions [2] of its toleranced proteins. The following ratio, which should ideally be close to one, is used to make a geometric assessment:

$$\overline{V}_\lambda(C) = Vol_\lambda(C)/Vol_{ref}(C). \tag{5.1}$$

Mining contacts. At the local level, the complexes encountered in the Hasse diagram can be used to evaluate protein contacts with respect to 3D templates known at atomic resolution. To quantitatively characterize pairwise contacts between instances of two protein types (P_i, P_j), we define a contact probability depending on the

[1] We track the evolution of connected components, but not that of higher order homology generators.

[2] In the bicolor setting, the volume of a red complex is defined from its constituting red restrictions in the CW Voronoi diagram. Practically, however, we add up the volumes of the restrictions in the power diagram, as explained in [CKL11].

stoechiometry k of the interaction between these two proteins by $p_{ij}^{(k)} = 1 - \lambda(P_i, P_j)/\lambda_{\max}$, with $\lambda(P_i, P_j)$ the first value of λ for which k contacts are established between instances of two protein types (P_i, P_j), and with $\lambda_{\max}$ the λ value where the growth process stops. As explained in Section 4.4, $\lambda_{\max}$ is set to retain acceptable volume ratios. The variation of $p_{ij}^{(k)}$ as a function of k, called the contact curve, is a key feature to assess whether an unambiguous stoichiometry exists for the contact between instances of two types. We use contact curves to define:

- (i) k_{high}, the largest stoichiometry observed for the probability $p_{ij}^{(1)}$;
- (ii) k_{low} as the largest stoichiometry for which $p_{ij}^{(k)} > 0$;
- (iii) k_{drop}, the stoichiometry maximizing the probability drop $\delta p_{ij}^{(k)} = p_{ij}^{(k)} - p_{ij}^{(k+1)}$;
- (iv) $s(k_{drop}) = p_{ij}^{(1)} / \delta p_{ij}^{(k_{high})}$ the significance of the largest variation with respect to $p_{ij}^{(1)}$.

Two prototypical contact curves illustrating these notions are presented on Fig. 5.2. The first one, for (Nup84, Nup145C), is the ideal case since $p_{ij}^{(k)}$ takes only two values: 1 or 0. The second one, that of (Nup84, Nup85) poses interpretation problems, since it does not contain any significant plateau.

5.2.3 Combining the Geometric, Topological and Biochemical Assessments

Assume that the red proteins are instances of types prescribed in a set T, typically corresponding to a TAP pulldown. The following parameters can be assessed.

Stoichiometry. Analyzing the complexes of the Hasse diagram has several interests: first, one sees whether the set T corresponds to a single complex or to a mixture of complexes; second, one can spot the isolated copies associated to the set T – see Section 5.2.2; third, if T corresponds to a TAP experiment, one can check whether each complex contains the tagged protein.

Symmetry. For an assembly with symmetries, one can compare the number of complexes with the expected number. For example, in the NPC, the multiplicity of selected complexes is expected to be 16.

Topological stability. In Section 5.2.2, the stability of a complex has been defined as the difference between its birth and death dates. This information is particularly relevant to know when a given complex collides with another one to form a larger complex. For an assembly involving a prescribed number of complexes, one expects the variation of the number of complexes as a function of λ to exhibit a plateau. Also, for an assembly with symmetries, the homogeneity of the model can be inferred from the stability of complexes featuring the same types, but located in different places.

Geometric accuracy. A complex may involve the correct protein instances, but may have a loose geometry. Comparing its volume to that occupied by its constituting instances is the goal of the volume ratio of Eq. (5.1).

Contact probabilities. A contact between two toleranced proteins is relevant if it appears early during the growth process. Comparing the contact probabilities of all contacts between toleranced proteins of two given types P_i and P_j gives: (i) an assessment on the significance of the interaction between protein instances of types P_i and P_j, and (ii) an estimation of the stoichiometry of the interaction between P_i and P_j.

5.3 Results: Contact Probabilities of All Pairs of Protein Types

5.3.1 Contact Probabilities versus Contact Frequencies

For $k = 1$, we compare the probability $p_{ij}^{(1)}$ to the contact frequency f_{ij}—refer to section 1.3.4 for the definition of f_{ij}. As discussed in Section 4.4, there are 29 protein types and 435 possible contacts, including homotipic ones.

Using the two probabilities $0 \leq a < b \leq 1$, the 435 contact frequencies f_{ij} are sorted into three classes in [ADV$^+$07a]:

$$F_1 : f_{ij} \leq a;\;\; F_2 : a < f_{ij} < b;\;\; F_3 : b \leq f_{ij}.$$

Similarly, we segregate the contacts observed from the Hasse diagram into the three classes

$$P_1^{(1)} : p_{ij}^{(1)} \leq a;\;\; P_2^{(1)} : a < p_{ij}^{(1)} < b;\;\; P_3^{(1)} : b \leq p_{ij}^{(1)}.$$

For $a = 0.25$, $b = 0.65$ and $\lambda_{\max} = 1$, the sizes of the classes are $\mid F_1 \mid = 325, \mid F_2 \mid = 79$ and $\mid F_3 \mid = 31$. Moreover, 93.5% of the contacts in F_3 belong to $P_3^{(1)}$, and 60.5% of the contacts in F_1 belong to $P_1^{(1)}$. The contact probability is more discriminative than the contact frequency since the maximum number of contacts in $P_2^{(1)}$ and F_2 are respectively of 53 and 79. For the more stringent values $a = 0.1$, $b = 0.9$ and $\lambda_{\max} = 1$, one has $\mid F_1 \mid = 220, \mid F_2 \mid = 196$ and $\mid F_3 \mid = 19$ contacts. Then, 79% of contacts in F_3 belong to $P_3^{(1)}$, 73% of contacts in F_1 belong to $P_1^{(1)}$, while the maximum number of contacts in $P_2^{(1)}$ is 95—to be compared to 196 contacts in F_2. See Figures 5.3 and 5.4.

We also use the values $a = 0.1$, $b = 0.9$ to report mismatches. A pair of types belonging to F_1 but $P_3^{(1)}$ is called *over-represented* in the toleranced model. Table 5.1 lists the 23 over-represented pairs. Note that all these contacts are over-represented for $\lambda_{\max} \geq 0.21$, which clearly indicates that the corresponding contacts appear early in the growth process. Similarly, a pair belonging to F_3 but $P_1^{(1)}$ is termed *under-represented* in the toleranced model. Table 5.2 lists such cases, which are under-represented for $\lambda_{\max} \leq 0.28$. The illustrations presented on Figures 5.5. and 5.6 clearly support our contact probability.

Contact	f_{ij}	$p_{ij}^{(1)}$	$\lambda_{\max}$
Nup59 Nup59	0	1	0
Pom34 Pom34	0.02	1	0
Nsp1 Nsp1	0.02	1	0
Nup60 Nup145N	0.03	1	0
Nup60 Pom34	0.03	1	0
Nup145N Nup49	0.04	1	0
Nup1 Nup145N	0.05	1	0
Nup60 Ndc1	0.06	1	0
Nup84 Nup60	0.07	1	0
Nsp1 Nup145N	0.07	1	0
Nup145C Nup60	0.08	1	0
Sec13 Nup159	0.08	1	0
Nsp1 Nup60	0.08	1	0
Nup49 Nup116	0.08	1	0
Nup57 Nup145N	0.08	1	0
Nsp1 Nup42	0.09	1	0
Nup60 Nup59	0.09	1	0
Nup42 Nup116	0.09	1	0
Nup57 Nup116	0.09	1	0
Sec13 Nup145N	0.1	1	0
Nup59 Pom34	0.03	0.9	0.15
Seh1 Nup60	0.06	0.9	0.18
Gle2 Nup57	0.08	0.9	0.21

Table 5.1: Over-represented pairs of types in the toleranced model for $a = 0.1$ and $b = 0.9$—that is pairs in $P_3^{(1)}$ and F_1. The last column is the smallest $\lambda_{\max}$ value for which the contact is over-represented—the smaller the value the more significant the contact.

Contacts	f_{ij}	$p_{ij}^{(1)}$	$\lambda_{\max}$
Nup192 Pom152	0.98	0	1
Nup170 Ndc1	0.91	0.1	0.35
Nup188 Nic96	1	0.1	0.32
Pom152 Pom34	1	0.1	0.28

Table 5.2: Under-represented pairs of types for $a = 0.1$ and $b = 0.9$, i.e. pairs in $P_1^{(1)}$ and F_3. The last column is the largest $\lambda_{\max}$ value for which the contact is under-represented—the larger the value the less significant the contact.

5.3.2 Contact Probabilities with Prescribed Stoichiometry k

To leverage the previous information, we now focus on pairs of types making a prescribed number of contacts. For $\lambda_{\max} = 1$, we observe 183 contacts (over 435) with $p_{ij}^{(1)} \geq 0.65$, but only 36 with $p_{ij}^{(16)} \geq 0.65$.

Inspecting all pairs. We inspect how the number of contacts such that $p_{ij}^{(k)} \geq 0.65$ decreases with k. For a given $\lambda_{\max}$ and two fixed probabilities $0 \leq a < b \leq 1$, the contacts observed in the Hasse diagram are partitioned into the classes $P_i^{(k)}, i = 1,2,3$. The variation of the cardinality of these classes with $\lambda_{\max}$ and k is displayed on Figure 5.7. We note that the curves are just shifted when $\lambda_{\max}$ varies, showing the consistency of contact probabilities with respect to $\lambda_{\max}$. In the following, we consider $\lambda_{\max} = 1$ (solid lines).
The red and blue curves show that the contact probability to have k instances of the contacts between two protein types decreases when k increases. The green curves show the discriminant property of the contact probability since less than 40 pairs of protein types are in $P_2^{(1)}$, and green curves tend to decrease when k increases.

Grouping proteins in sub-complexes. Consider the graph G whose edges correspond to types displaying at least 11 contacts. Term a sub-graph H of G a *complete* sub-graph if there exists an edge between every pair of nodes of H. In addition, a sub-graph H of G is termed a *quasi-complex* sub-graph if by adding one edge, it becomes a complete sub-graph. Computing the cliques and quasi-cliques of size four, as seen from Fig. 5.8, uncovers sub-complexes of the NPC, including two sub-units of the Y-complex (one containing Nup120, Nup85 and Seh1, the other one containing Nup133, Nup84, Nup145C and Sec13), the T-complex and the Nup82-complex.

5.4 Results: Y-complex Analysis

5.4.1 Contact probabilities

We have seen in Section 4.2.1 that the current model proposed for the Y-complex involves six contacts between the seven protein instances.
At the local level, previously established contact frequencies f_{ij} between the various types present in the Y-complex were not discriminative [ADV$^+$07a], see Table 5.3. In contrast, contact probability analysis revealed that out of the six expected binary contacts within the NPC, four had a high probability to occur 16 times as expected ($k_{drop} = 16$) and one was slightly less consistent ($k_{drop} = 12$). However only two contacts was observed between Nup120 and Nup145C whereas additional pairs had an unexpected high contact probability, indicating that these proteins are poorly positioned with respect to each other in the current model. While [SMD$^+$09] previously suggested that interaction between Nup133 and Nup120 was required for ring closure, contact analysis only revealed 1 significant contact between these two proteins with however 6 additional contacts between Nup133 and Nup85.

5.4.2 Stoichiometry, symmetry, stability

We have described in section 4.2.1 the two competing models for the embedding of the Y-complex in the NPC. These two models imply that different copies of the Y-complex interact, and question the prominence of contacts within a Y-complex, and in-between Y-complexes.

Protein types	f_{ij}	k_{high}	k_{drop}	$p_{ij}^{(k_{drop})}$	$s(k_{drop})$	$\min \overline{V}_{\lambda_{k_{high}}}$	$\max \overline{V}_{\lambda_{k_{drop}}}$
(Nup133, Nup84)	0.571	16	16	1.00	1.00	0.77	0.91
(Nup145C, Nup84)	1.000	16	16	1.00	1.00	0.77	0.88
(Nup120, Seh1)	0.837	16	16	1.00	1.00	0.82	0.92
(Nup133, Nup145C)	0.589	16	16	1.00	1.00	0.82	0.90
(Nup120, Nup85)	0.569	16	16	1.00	1.00	0.88	0.98
(Nup85, Seh1)	1.000	12	16	0.93	1.07	0.77	1.27
(Nup120, Sec13)	0.284	5	16	0.64	1.56	0.82	4.10
(Nup133, Sec13)	0.381	11	14	0.69	1.45	0.80	3.91
(Nup84, Sec13)	0.66	8	14	0.54	1.85	0.70	4.49
(Nup85, Sec13)	0.227	4	13	0.57	1.76	0.77	8.57
(Nup145C, Sec13)	0.503	12	12	1.00	1.00	0.79	0.86
(Sec13, Seh1)	0.233	4	9	0.65	1.55	0.57	6.88
(Nup120, Nup84)	0.487	1	8	0.60	1.68	0.91	3.26
(Nup84, Seh1)	0.376	1	7	0.49	2.06	0.79	3.89
(Nup133, Nup85)	0.478	1	6	0.79	2.34	1.28	2.65
(Nup145C, Seh1)	0.359	1	4	0.34	2.98	2.19	3.04
(Nup120, Nup145C)	0.498	1	2	0.86	2.21	1.02	1.49
(Nup120, Nup133)	0.465	1	1	1.00	2.18	0.92	0.92
(Nup84, Nup85)	0.543	1	1	1.00	2.71	0.89	0.89

Table 5.3: Contact probabilities versus contact frequencies for the Y-complex. Out of 21 pairs of the 7 protein types, 19 pair yield at least one binary complex—pairs with no contact in the TOM are not represented. The grey column displays the contact frequencies f_{ij} of [ADV$^+$07a]. Pairs are sorted by decreasing k_{drop}, and are color-coded as follows: green: contacts of the skeleton of the Y-complex; red: putative contact accounting for the closure of the two rings [SMD$^+$09]; orange: predominant contact accounting for the closure of the two rings in the TOM.

The evolution of complexes involving the seven types of the Y-complex is provided by the Hasse diagram on Figure 5.9 (Top). Out of 16 expected copies of the Y-complex, 11 are observed in the range $\lambda = 0$ ($\overline{V}_\lambda = 0.86$) and $\lambda = 0.31$ ($\overline{V}_\lambda = 2.14$). These correspond to the green nodes on the Hasse diagram, one of them being singled out on Figure 5.9 (Bottom-left). The stability of these 11 complexes is heterogeneous as their lifetimes span the range $l(C) = 0.01$ ($\Delta\overline{V}_\lambda = 0.06$) and $l(C) = 0.44$ ($\Delta\overline{V}_\lambda = 2.47$). Also, they do not coexists since the intersection of their lifetime intervals is empty. These observations show that contacts between protein instances belonging to several copies of the Y-complex can prevail over contacts within the isolated copies.

5.4.3 Further In-Silico Experiments

On the closure of the two rings of the NPC. One of the models described in Section 4.2.1 supports the formation of two rings made of eight Y-complexes each.
Here, we investigate the implication of Nup133 in the ring closure. We establish the role of Nup133 by painting it in blue, and by observing that one gets six instead of two connected components, see Figure 5.10. However, while it had previously been suggested that the interaction between Nup133 and Nup120 was mandatory for the ring closure [SMD$^+$09], contact analysis only reveals 1 significant contact between these two proteins, with however 6 additional contacts between Nup133 and Nup85.

On the connectedness of copies of the Y-complex. As shown on Figure 5.11, each copy of the Y-complex is split into two components. That is, for each copy, there exists a value of the probability such that the level set surfaces of the maps restricted to this copy have two connected components, one including Nup145C and the other one Y_X-short-arm. The graph of contacts of the Y-complex on Figure 5.8 clearly supports the split of the Y-complex into two sub-units. Furthermore, the contact probabilities on Table 5.3 show that (i) there are 16 contacts between Nup133, Nup84 and Nup145C with a probability $p_{ij}^{(k_{drop})} = 1$; (ii) there are 16 contacts between Nup85 and Nup120 with a probability $p_{ij}^{(k_{drop})} = 1$; (iii) except for one contact between Nup84 and Nup85, there is no other contact involving two of the five protein types with $p_{ij}^{(k_{drop})} = 1$. This results confirm the split of the Y-complex into two sub-units observed on the density maps.

5.5 Results: T-complex Analysis

5.5.1 Contact probabilities

We have seen in Section 4.2.2 that the T-complex involves at least three contacts between the four protein instances.
The contact probabilities related to the T-complex are summarized in Table 5.4. Notice that Nsp1-1 and Nic96-1 do not make contacts with Nup49 or Nup57, strengthening the fact that Nsp1-2 and Nic96-2 likely represent the populations of Nsp1 and Nic96 contributing to the T-complex. Note that unlike anticipated [SSF$^+$08], only one contact is observed in the TOM between Nic96-2 and Nup57.

5.5.2 Stoichiometry, symmetry, stability

We have seen in Section 4.2.2 that the 16 copies of the T-complex are embedded symmetrically in the NPC, without contact between different copies.
The bottom right Hasse diagram at Figure 5.12 shows that the 16 copies—the expected number—of the T-complex get formed thanks to merges in-between $\lambda = 0$ ($\overline{V}_\lambda = 0.72$) and $\lambda = 0.15$ ($\overline{V}_\lambda = 1.24$). Their lifetimes are rather homogeneous since they vary in-between $l(C) = 0.10$ ($\Delta\overline{V}_\lambda = 0$) and $l(C) = 0.33$ ($\Delta\overline{V}_\lambda = 1.29$), and the copies coexist in-between $\lambda = 0.15$ and $\lambda = 0.22$. These results show that contacts inside a copy of the T-complex prevail over contacts between different copies of the T-complex.

5.5.3 Further In-Silico Experiments

Selecting ambiguous density maps. We have seen in Section 4.3 that there are 2 density maps for Nic96 (and Nsp1), each involving 16 instances. On the other hand, the stoichiometry of the T-complex is 16, which requires

Protein types	$f_{ij}{}^{\star}$	k_{high}	k_{drop}	$p_{ij}^{k_{drop}}$	$s(k_{drop})$	$\min \overline{V}_{\lambda_{k_{high}}}$	$\max \overline{V}_{\lambda_{k_{drop}}}$
(Nup49, Nup57)	1	16	16	1.00	1.00	0.47	0.76
(Nsp1-2, Nup49)	1	16	16	1.00	1.00	0.65	0.83
(Nsp1-2, Nup57)	1	16	16	1.00	1.00	0.67	0.86
(Nic96-2, Nsp1-2)	1	6	15	0.86	1.16	0.77	1.82
(Nic96-1, Nsp1-1)	1	7	15	0.82	1.22	0.83	3.66
(Nsp1-1, Nsp1-2)	0.021	1	15	0.63	1.58	0.99	4.63
(Nic96-1, Nsp1-2)	1	6	11	0.93	1.08	0.77	1.51
(Nic96-2, Nup49)	0.442	1	10	0.85	1.18	0.90	2.20
(Nup57, Nup57)	0.005	1	8	0.71	1.41	1.93	2.74
(Nsp1-1, Nsp1-1)	0.021	2	7	0.27	3.72	0.88	8.98
(Nic96-2, Nup57)	0.424	1	3	0.75	1.33	1.22	2.08

Table 5.4: Contact probabilities versus contact frequencies for the T-complex. Pairs with no contact in the TOM are not represented. The grey column displays the contact frequencies f_{ij} of [ADV^{+}07a]. For contact frequencies (grey column), the * denotes the fact that these frequencies did not discriminate between twin maps, i.e. Nps1-1 and Nsp1-2 on the one hand, and Nic96-1 and Nic96-2 on the other hand. The green and orange rows correspond to the six pairs involved in the T-complex with Nsp1-2 and Nic96-2.

16 instances of Nic96 and Nsp1.

Since each map contains one protein instance per half-spoke of the NPC, out of the four possible pairs, (two options for Nic96 and two options for Nsp1) we select the pair producing the best results i.e maximizing global results. The four resulting Hasse diagrams are shown on Figure 5.12. Only the Hasse diagram at the Bottom Right reveals 16 isolated copies of the T-complex, motivating the selection of the corresponding two density maps. Note that a calculation with the four density maps would have required selecting the relevant instances of Nic96 and Nsp1 within each half-spoke, an ill-posed problem.

5.6 Results: Nup82-complex Analysis

5.6.1 Contact probabilities

The Nup82-complex, as discussed in Section 4.2.3, involves at least two contacts between the three protein instances.

Contact probabilities related to the Nup82-complex are summarized on Table 5.5. We first note that we observe eight instances of the homo-dimer (Nup82-1, Nup82-2) with $p_{ij}^{(8)} = 1$. Considering this homo-dimer as the central piece of the Nup82-complex, we observe that eight instances of Nsp1-2 and eight instances of Nup159 are in contact with the homo-dimer with $p_{ij}^{(8)} \geq 0.84$. Note that the contact frequencies do not allow to discriminate the sub-populations of Nsp1 and Nup82 that are in contact.

5.6.2 Stoichiometry, symmetry, stability

As done for the Y-complex and the T-complex, we inspect the stability of contacts inter- and intra- copies of the Nup82-complex.

The right Hasse diagram at Figure 5.13 shows that the 8 copies—the expected number—of the Nup82-complex get formed thanks to merges at $\lambda = 0$ ($\overline{V}_{\lambda} \leq 0.81$). Their lifetimes are totally homogeneous since all the copies coexist in-between $\lambda = 0$ and $\lambda = 1$.

5.6.3 Further In-Silico Experiments

Selecting ambiguous density maps. Remind that there are only eight instances of the Nup82-complex located on the cytoplasmic side.

For contact frequencies (grey column), the * denotes the fact that these frequencies did not discriminate between twin maps, i.e. Nps1-1 and Nsp1-2 on the one hand, and Nup82-1 and Nup82-2 on the other hand.

Protein types	f_{ij} ⋆	k_{high}	k_{drop}	$p_{ij}^{k_{drop}}$	$s(k_{drop})$	$\min \overline{V}_{\lambda k_{high}}$	$\max \overline{V}_{\lambda k_{drop}}$
(Nup159, Nup82-1)	0.951	8	8	1.00	1.00	0.68	0.82
(Nup159, Nup82-2)	0.951	8	8	1.00	1.00	0.65	0.80
(Nup82-1, Nup82-2)	0.284	8	8	1.00	1.00	0.44	0.68
(Nsp1-2, Nup82-2)	1	6	8	0.92	1.09	0.60	0.90
(Nsp1-1, Nup82-2)	1	6	8	0.96	1.04	0.67	0.89
(Nsp1-2, Nup82-1)	1	4	8	0.60	1.66	0.66	2.26
(Nsp1-1, Nup82-1)	1	1	5	0.72	1.39	0.83	3.68
(Nsp1-2, Nup159)	0.187	2	5	0.93	1.08	0.81	1.09
(Nsp1-1, Nup159)	0.187	1	1	0.97	1.08	0.98	0.98

Table 5.5: Contact frequencies f_{ij} from [ADV^{+}07a] and contact probabilities derived from the TOM for all possible pairs of protein types of the Nup82-complex (Nup82, Nsp1, Nup159).

Since Nsp1 is divided in two sub-populations, the there are two possibilities of composition for the Nup82-complex: with Nsp1-1 or Nsp1-2. Note that Nup82 is divided in two sub-populations too: each instance of the Nup82-complex contains an instance of the homo-dimer (Nup82-1,Nup82-2). We computed the two possible Hasse diagrams corresponding to the different possible Nup82-complex and compare them in Figure 5.13. Note that while distinct fractions of Nsp1 are expected to interact with the T-complex and the Nsp1-Nup82-Nup159 containing complex respectively [BBH01], Hasse diagrams indicate that, as also observed for the T-complex, Nsp1-2 but not Nsp1-1 leads to the formation of the expected eight isolated copies of this complex.

5.7 Artwork

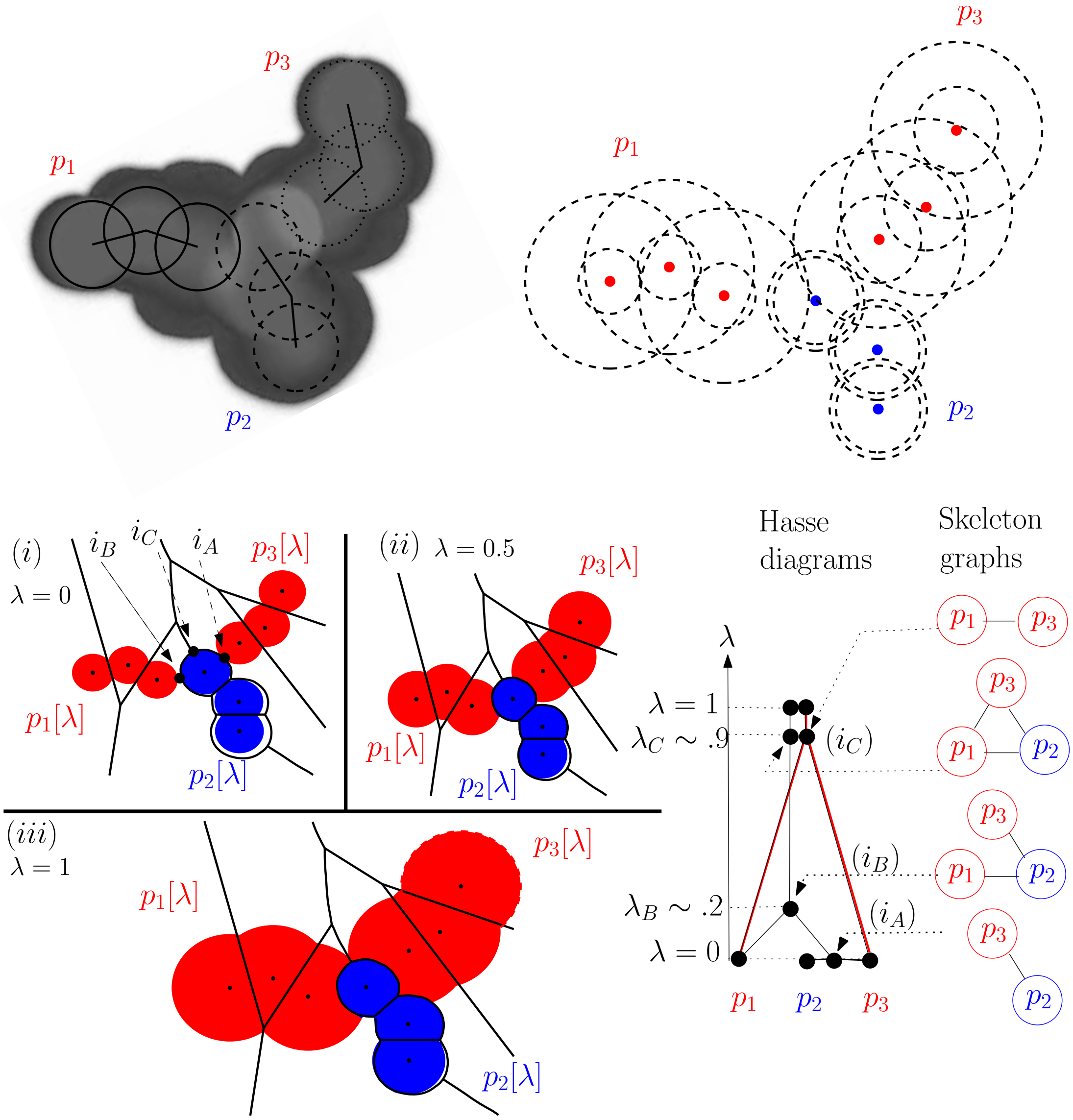


Figure 5.1: Tracking the interactions of three toleranced proteins of three toleranced balls each. **(Top left)** Three conformations of three flexible molecules, and a probability density map whose color indicates the probability of a given point to be covered by a random conformation of the ternary complex — from low (black pixels) to high (gray pixels) probabilities. **(Top right)** The associated bicolor toleranced model, with one blue and two red molecules. Each toleranced molecule consists of a set of pairs of concentric balls, the inner and outer balls. **(Bottom left)** Sub-figures (i,ii,iii) respectively show grown balls $\overline{B_i}[\lambda]$ for $\lambda = 0, 0.5, 1$. The region of the plane consisting of points first reached by a growing toleranced ball is the Voronoi region of this ball, represented by solid lines. Colored solid regions feature the *restrictions* i.e. the intersection of a growing ball and its Voronoi region. Along the growth process, the restrictions intersect in three points i_A, i_B, i_C, represented as black points. **(Bottom right)** Hasse diagrams encoding contacts between the protein instances. Black tree: all instances; red tree: red instances only.

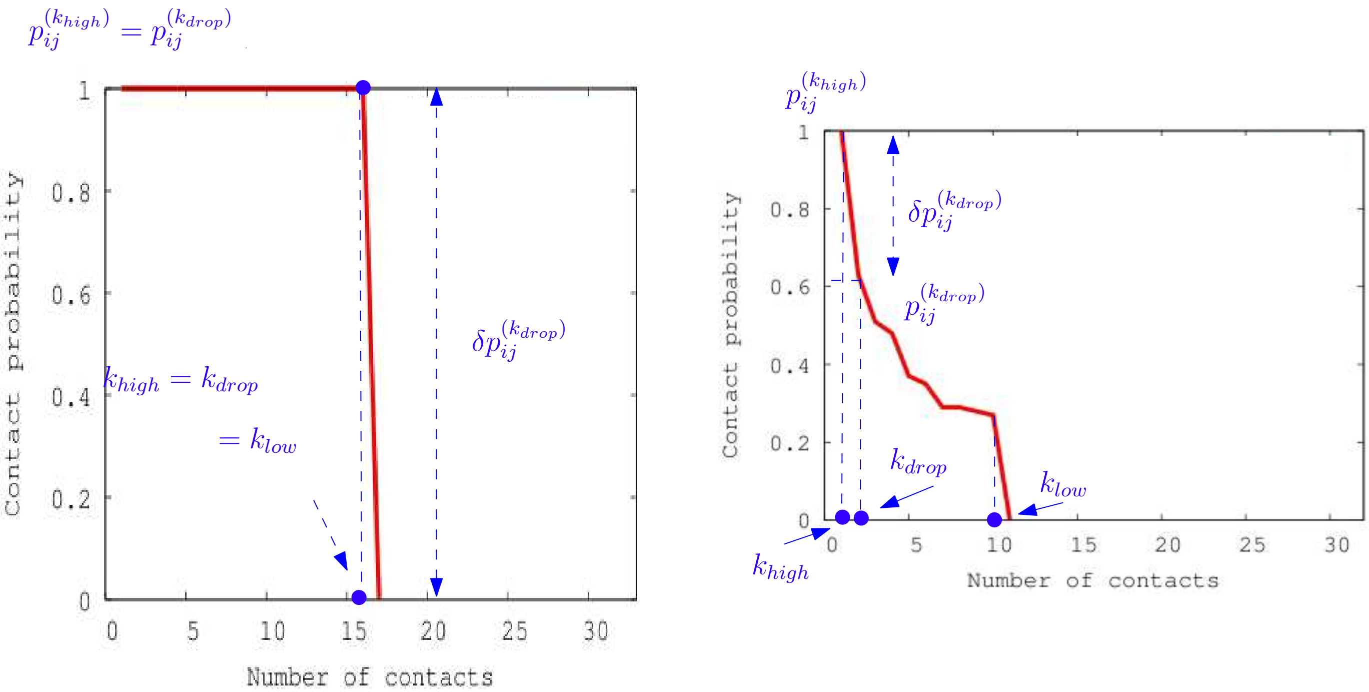


Figure 5.2: Example of two contact curves related to the Y-complex. **Left.** Contact curve of (Nup84, Nup145C): 16 contacts are observed at $\lambda = 0$, and the contact probability is null for $k = 17$, which is the ideal situation since both types have a stoichiometry of 16. With a value of one, the significance coefficient $s(k_{drop})$ is also perfect. **Right.** Contact curve of (Nup84, Nup85) The value $s(k_{drop}) = 2.71$ shows that the largest probability drop is not significant with respect to $p_{ij}^{(1)}$. In short, it is not possible to unambiguously choose a stoichiometry k for these two types.

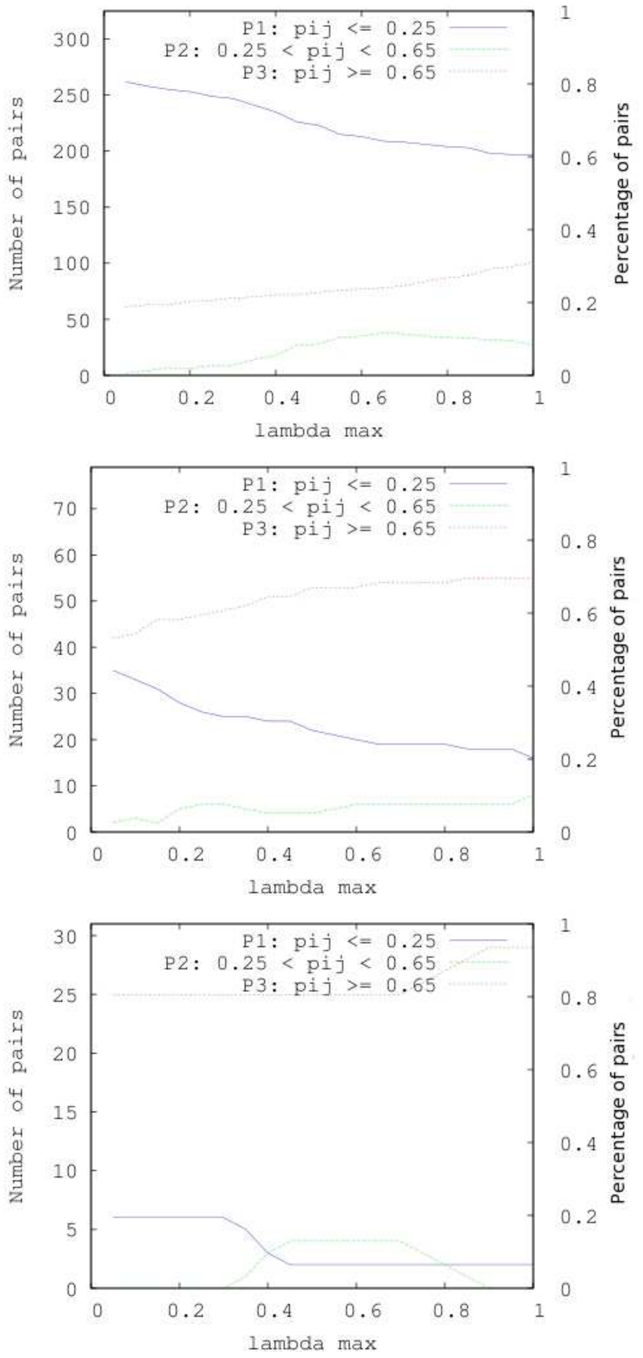


Figure 5.3: Partitioning the pairs of protein types into three classes reveal that contact probabilities are more discriminatory than contact frequencies from [ADV$^+$07a]. The three classes in each picture are $P_i^{(1)}, i = 1,2,3$, represented respectively by a solid blue curve ($P_1^{(1)}$), a dashed green curve ($P_2^{(1)}$) and dotted red curve ($P_3^{(1)}$). The pairs of protein types were separated following their contact frequency f_{ij}, from [ADV$^+$07a]. **Top:** low frequencies F_1 i.e. $f_{ij} \leq a$; **Middle:** medium frequencies F_2 i.e. $a < f_{ij} < b$; **Bottom:** high frequencies F_3 i.e. $b \leq f_{ij}$. Following [ADV$^+$07a], the thresholds are $a = 0.25$ and $b = 0.65$.

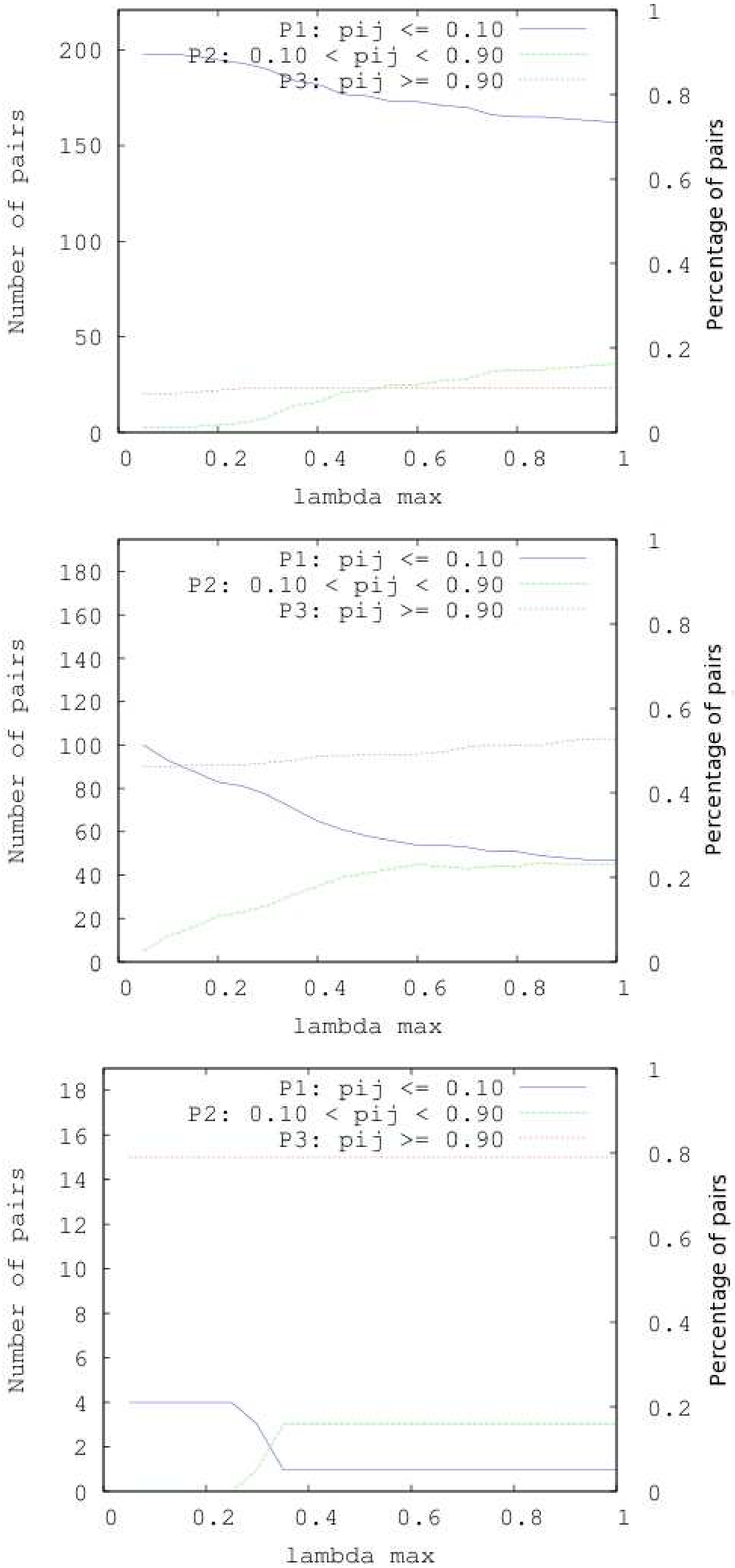


Figure 5.4: Partitioning the pairs of protein types into three classes reveal that contact probabilities are more discriminatory than contact frequencies from [ADV$^+$07a]. The three classes in each picture are $P_i^{(1)}, i = 1,2,3$, represented respectively by a solid blue curve ($P_1^{(1)}$), a dashed green curve ($P_2^{(1)}$) and dotted red curve ($P_3^{(1)}$). The pairs of protein types were separated following their contact frequency f_{ij}, from [ADV$^+$07a]. **Top:** low frequencies F_1 i.e. $f_{ij} \leq a$; **Middle:** medium frequencies F_2 i.e. $a < f_{ij} < b$; **Bottom:** high frequencies F_3 i.e. $b \leq f_{ij}$. Thresholds are $a = 0.1$ and $b = 0.9$.

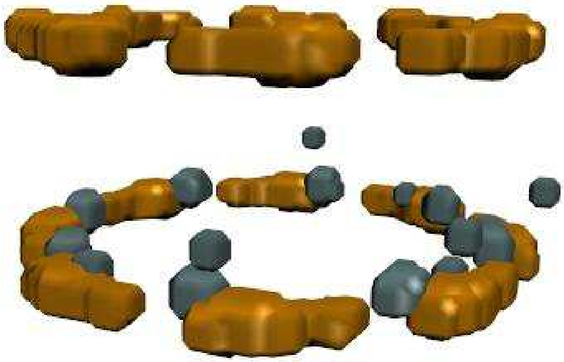


Figure 5.5: An example of over-represented pair in the toleranced model. The overlapping density maps of Nup84 (stoichiometry: 16) and Nup60 (stoichiometry: 8), from `http://salilab.org/npc/`, visualized with VMD. Their contact frequency from [ADV$^+$07a] is $f_{ij} = 0.07$, while the contact probability from the toleranced model is $p_{ij}^{(1)} = 1$.

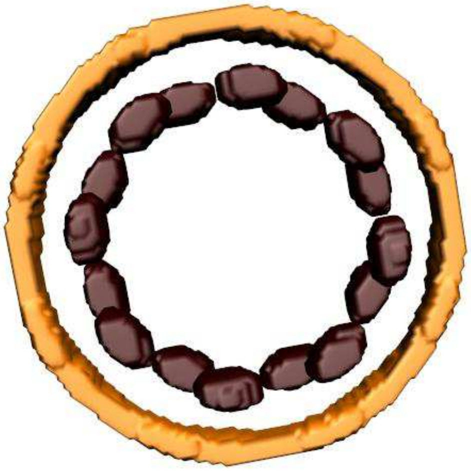


Figure 5.6: An example of under-represented pair in the toleranced model. The disjoint density maps of Nup192 (stoichiometry: 16) and Pom152 (stoichiometry: 16), from `http://salilab.org/npc/`, visualized with VMD. Their contact frequency from [ADV$^+$07a] is $f_{ij} = 0.98$, while the contact probability from the toleranced model is $p_{ij}^{(1)} = 0$.

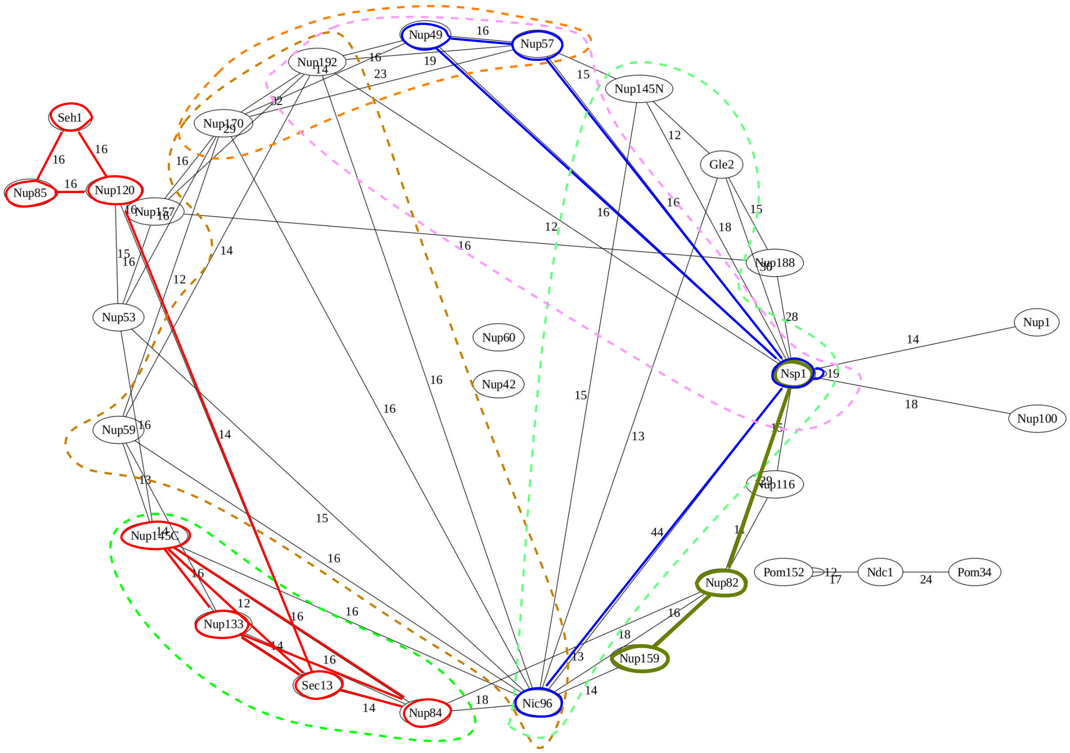


Figure 5.8: Computing cliques and quasi-cliques in the graph of contacts with prescribed stoichiometry identifies sub-systems of the NPC. The graph only involves edges corresponding to pairs of protein types (P_i, P_j) such that $p_{ij}^{(k>10)} \geq 0.65$ for $\lambda_{\max} = 1$. The red, blue and dark green sub-graphs respectively correspond to the Y-complex, T-complex and Nup82-complex. The nodes contained in each of the five dashed regions define a complete sub-graph i.e. a clique of size four.

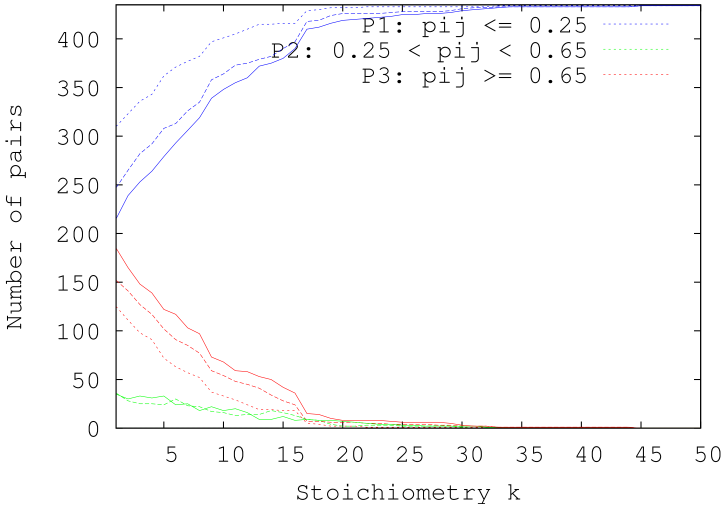


Figure 5.7: Evolution of low, medium and high contact probabilities $p_{ij}^{(k)}$ as a function of the stoichiometry of contacts between the types P_i and P_j. The contact probabilities are grouped in the three classes $P_1^{(k)}$ (blue curve), $P_2^{(k)}$ (green curve) and $P_3^{(k)}$ (red curve). **Dotted lines** for $\lambda_{\max} = 0$, **dashed lines** for $\lambda_{\max} = 0.5$ and **solid lines** for $\lambda_{\max} = 1$. Following [ADV$^+$07a], the thresholds $a = 0.25$ and $b = 0.65$ have been used.

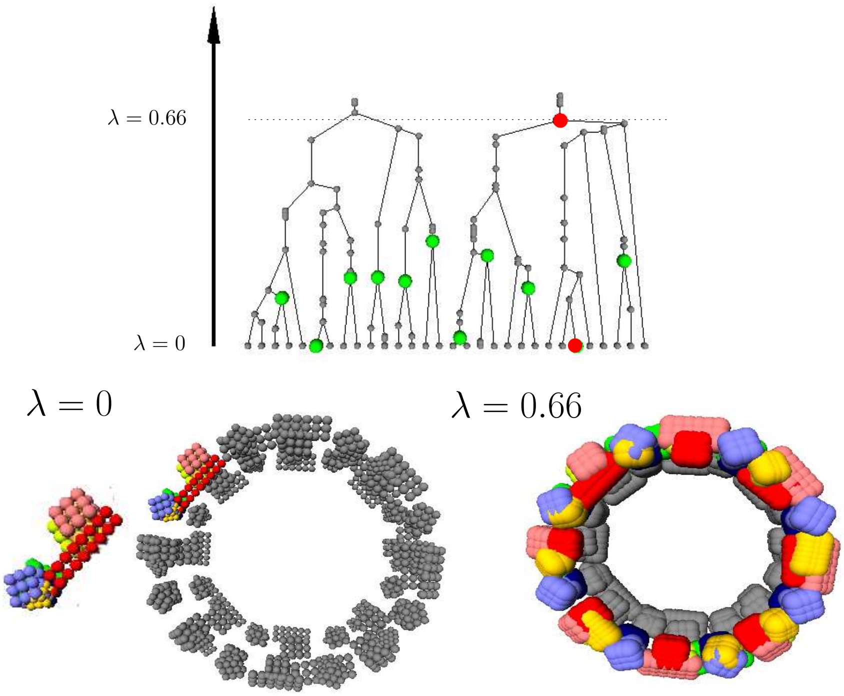


Figure 5.9: The Hasse diagram of the Y-complex reveals 11 isolated copies, and evidences the closure of the two rings involving each 8 copies of the Y-complex. **Top.** Hasse diagrams of the Y-complex with its seven types painted in red. The green nodes correspond to isolated copies, and the two red ones correspond to the colored complexes represented on the bottom. **Bottom.** Snapshot of the toleranced model at $\lambda = 0$ **(bottom left)**, with an isolated copy shown as inset, and at $\lambda = 0.66$ **(bottom right)**, when the upper ring appears. The protein instances highlighted with the color code of Figure 4.1 correspond to the complexes in the red fat nodes of the Hasse diagram (top)

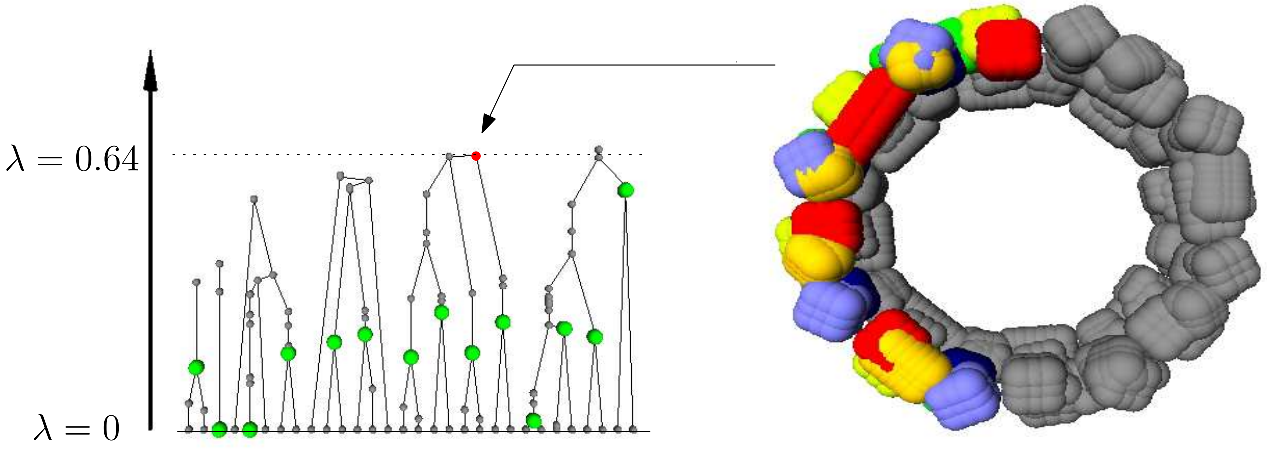


Figure 5.10: Painting Nup133 in blue evidences its role in the closure of the two rings. **Top.** The Hasse diagram with Nup133 painted in blue involves six connected components as opposed to two—compare with Fig. 5.9. **Bottom.** The corresponding toleranced model at $\lambda = 0.64$. The colored complex corresponds to the red node in the Hasse diagram.

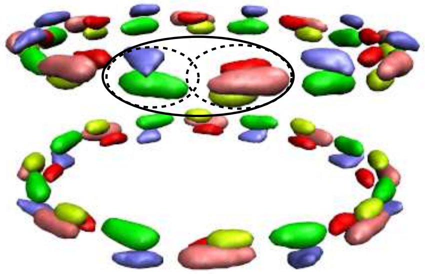


Figure 5.11: All copies of the Y-complex are split in two pieces in the probability density maps of [ADV$^+$07a]. The union of the level-set surfaces from the probability density maps of protein types of Y-main are shown—the intensity used to contour a map is half of the maximum value observed for that map. The color codes are those of Figure 4.1. The circled region illustrates the split of a Y-complex into two pieces (Nup133, Nup84, Nup145C) and (Nup120, Nup85).

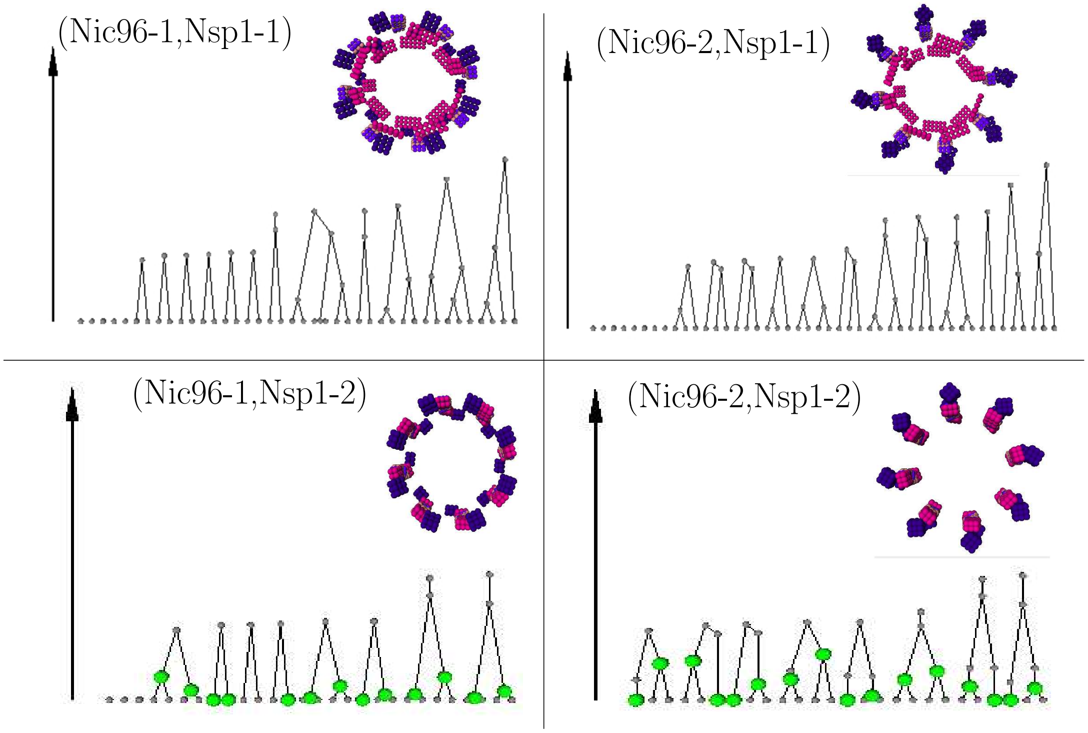
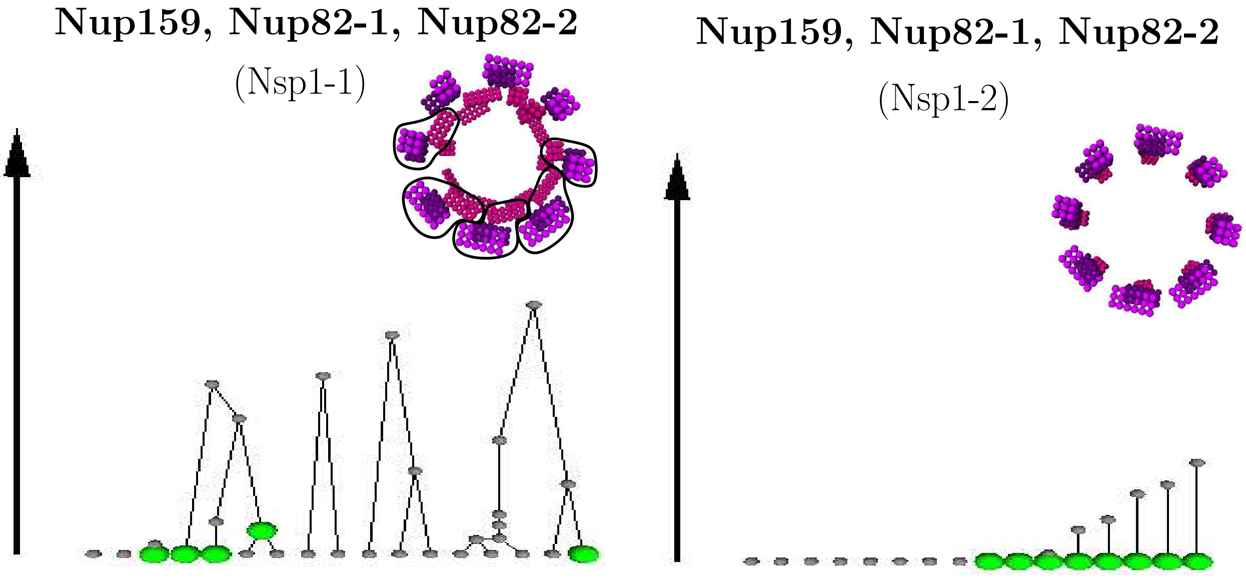


Figure 5.12: The Hasse diagrams of the T-complex reveal that the Nic96-2 and Nsp1-2 sub-populations form privileged contacts. The four Hasse diagrams associated to the four density maps of Nic96 and Nsp1 are shown—these two protein types define the T-core of the T-complex. Green nodes correspond to isolated copies. The associated toleranced models in each case are shown as inset. The two protein types selected for our study are those of the Bottom-Right Figure.



Nup159, Nup82-1, Nup82-2

(Nsp1-1)

Nup159, Nup82-1, Nup82-2

(Nsp1-2)

Figure 5.13: The Hasse diagrams of the Nup82-complex reveal that the same instances of Nsp1 are used in the T-complex and the Nup82 complex. The two possible hasse diagrams of Nup82-complex and their associated toleranced model are shown: with Nsp1-1 on the **left** and with Nsp1-2 on the **right**. The fat green nodes on the Hasse diagrams are the isolated copies. The corresponding protein complexes on the toleranced model are in-circles for the left case.

5.8 Supplemental

5.8.1 Partial Computation of the λ-Complex

λ-complex versus partial λ-complex. As discussed in Section 5.2.1, depending on the number of its constituting toleranced balls, a system may be investigated using the λ-complex or the partial λ-complex. Both algorithms were developed in C++, as discussed in Chapter 7, and the software was run on a dual core Intel Extreme CPU X7900 2.80GHz with RAM size of 8Go, under Fedora Core 14. The following running times were observed:

- The computation of the λ-complex on a complete half-spoke took about 15 hours. This computation for the whole NPC model halted after 6 days, due to a memory allocation failure.

- The computation of the partial λ-complex of the half-spoke and full NPC model respectively took less than one second and about 1 minute.

To assess the incidence of using the partial λ-complex rather then the λ-complex, recall that a contact between two proteins p_1 and p_2 corresponds to an edge of the λ-complex involving one ball of p_1 and one ball of p_2, and that such an edge has a status (Gabriel or not Gabriel). Using the partial complex may yield one of the following two discrepancies between p_1 and p_2 (Figure 5.14):

- No edge connecting balls of p_1 and p_2 is Gabriel. In that case, the connexion between p_1 and p_2 is absent from the Hasse diagram derived from the partial λ-complex.

- There is at least one edge which is Gabriel, but this edge is encountered at $\lambda_G > \lambda_{NG}$, with λ_{NG} the value of λ corresponding to the first non Gabriel edge between balls of the two proteins. In that case, p_1 and p_2 are connected in the Hasse diagram derived from the partial λ-complex, but at λ_G.

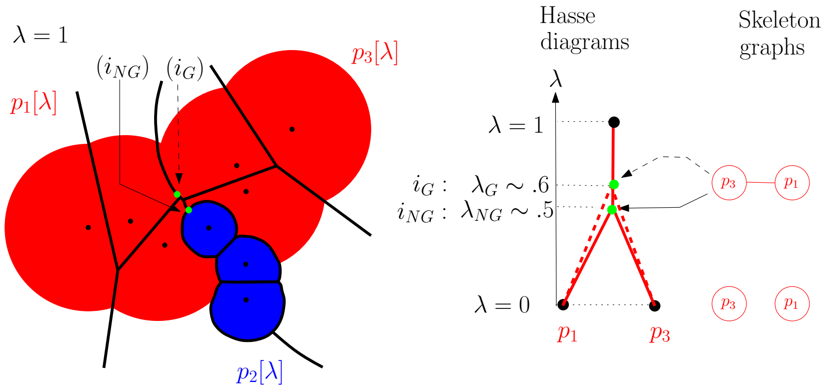


Figure 5.14: λ-complex versus partial λ-complex. **Left.** A toleranced model of three proteins p_1, p_2, p_3 instantiated at $\lambda = 1$. Its associated compoundly weighted Voronoi diagram is drawn in solid black lines. The green points correspond to the first non Gabriel (i_{NG}) and first Gabriel (i_G) contact between the red proteins (p_1, p_3). **Right.** The Hasse diagram from the λ-complex (red solid lines) and the partial λ-complex (red dashed lines) restricted to the red proteins (p_1, p_3). For the λ-complex, the red proteins are connected at $\lambda_{NG} \sim 0.5$ at (i_{NG}). For the partial λ-complex, the red proteins are connected at $\lambda_G \sim 0.6$ at (i_G).

Missed protein contacts: global snalysis. Table 5.6 compares the number of edges and contacts for a half-spoke, from which it is seen that using the partial λ-complex yields a decrease of the number of edges and contacts of 62% and 31% percents, respectively. Moreover, as shown on Figure 5.15, the number of missed contacts increases slowly when λ increases, with only eight missing protein contacts for $\lambda < 0.5$.

	# edges	# contacts
Whole λ-complex	5947	193
Partial λ-complex	2227	133

Table 5.6: λ-complex versus partial λ-complex: comparison of the number of edges connecting toleranced balls, and of the number of contacts between protein instances.

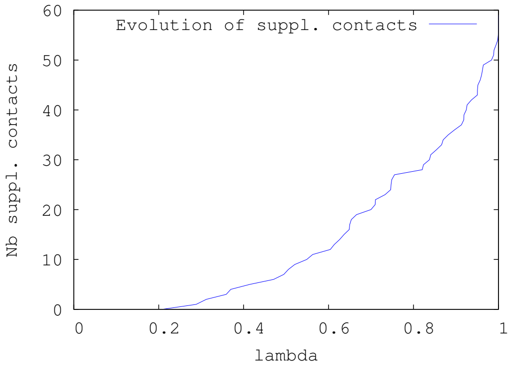


Figure 5.15: Evolution of the number of missed contacts between protein instances when using the partial λ-complex, as a function of λ.

Missed protein contacts: sub-complexes analyzed in this study. Regarding the protein contacts involved in the Y-complex, three differences between the two computations are observed at the level of one half-spoke :
– one contact (Nup85, Sec13) appears earlier, namely at $\lambda_{NG} = 0.39$ instead of $\lambda_G = 0.44$.
– two contacts are missed in the partial λ-complex: (Nup84, Seh1) at $\lambda = 0.88$ and (Nup120, Nup84) at $\lambda = 0.90$. Since the closure of the two rings is done at $\lambda = 0.64$, these two events are not relevant.
Concerning the T-complex and the Nup82-complex, there is no difference between protein contacts in both computations.

To conclude, using the partial λ-complex has no incidence on the results presented in this study, and makes the calculations tractable.

Chapter 6

Probing a Continuum of Macro-molecular Assembly Models with Graph Templates of Sub-complexes

6.1 Introduction - Rationale

Growing a toleranced model yields merges of complexes. Each such complex corresponds to a set of protein instances forming a connected domain, and the contacts between these instances can be represented by a graph. In this chapter, we aim at comparing this graph against a template, that is a model graph encoding the pairwise contacts between the same protein instances.
We first introduce in Section 6.2 theory and tools for comparing a protein complex of the Hasse diagram to a given template. Then, we assess templates defined in Chapter 4, for the Y-complex in Section 6.3, and for the T-complex in Section 6.4 with respect to the Hasse diagram computed in Chapter 5.

6.2 Analysis Tools for Protein Complex in a Hasse Diagram

6.2.1 Comparing a Protein Complex to a Template

Search of protein complexes similar to a template. From a topological standpoint, a protein complex C associated to a node of the Hasse diagram is characterized by its skeleton graph, see Figure 5.1. We want to compare the skeleton graph of a complex C against that of a template T of C. Practically, T shall be a co-crystallized complex or a high-resolution model built in-silico, and the protein types in T identify the red proteins of the bicolor setting. We formalize this comparison in terms of graph theory.

The skeleton graph G_C corresponds to a complex C whose nodes are protein instances i.e. each instance carries a unique identifying label.
On the one hand, the nodes of G_t are protein types, so that a node of G_C (a protein instance) can be uniquely mapped to a node of G_t (a protein type). This latter assumption is warranted by the fact that the templates of the NPC to be analyzed have at most one instance of each protein type, since we actually deal with isolated copies.
We assume that all the types of the instances present in the protein complex C are present in the template skeleton graph T. But the complex C may not feature instances of all the types found in the template T. We therefore denote $G_{T|C}$ the *restricted template* i.e. the graph obtained by removing from G_t all the nodes whose protein types are not found in the protein instances of G_C, and the edges incident on these nodes. To compare the graphs $G_{T|C}$ and G_C, we use the concept of *matching*.

Maximal Common Sub-graphs. Computing matchings is tantamount to computing maximal cliques [CK05], and of particular interests are the matchings associated to the so-called Maximal Common Induced Sub-graph (MCIS) and Maximal Common Edge Sub-graph (MCES) of $G_{T|C}$ and G_C.

Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two undirected labelled graphs.

Definition. 6. *A* **Maximal Common Edge Sub-graph (MCES)** *of G_1 and G_2 is a graph H that is isomorphic to sub-graphs G'_1 of G_1 and G'_2 of G_2, such that there is no other Common Edge Sub-graph H' of G_1 and G_2 containing H.*

Definition. 7. *An* **induced sub-graph** *G' of G is a sub-graph of G such that for all pairs of vertices (u,v) of G', (u,v) is an edge of G' iff it is an edge of G.*

Definition. 8. *A* **Maximal Common Induced Sub-graph (MCIS)** *of G_1 and G_2 is a graph H that is isomorphic to* induced sub-graphs *G'_1 of G_1 and G'_2 of G_2, such that there is no other Common Induced Sub-graph H' of G_1 and G_2 containing H.*

These notions are illustrated on Figure 6.1. Notice in particular that a MCES or MCIS calculation yields in general several matchings.

6.2.2 Analyzing Perfect and Alternate Matching

Signature of a matching. A *matching* from $G_{T|C}$ to C maps vertices of $G_{T|C}$ (protein types of the template) to vertices of G_C (protein instances of the complex), and edges of $G_{T|C}$ (contacts within the template) to edges of G_t (contacts within the complex). Taking the template as reference, we assess a matching with its signature, illustrated on Figure 6.2:

- *Matching protein type(s):* a protein type of $G_{T|C}$ with a corresponding instance in C. This set is denoted $V^{\sim}$.
- *Missing protein type(s):* a protein type of $G_{T|C}$ with no corresponding instance in C. This set is denoted V^{-}.
- *Matching contact(s):* a contact in $G_{T|C}$ with a counterpart in C. This set is denoted $E^{\sim}$.
- *Missing contact(s):* a contact in $G_{T|C}$ with no counterpart in C. This set is denoted E^{-}.
- *Extra contact(s):* a contact in C with no counterpart in $G_{T|C}$. This set is denoted E^{+}.

Using these sets, the *signature* of the matching A is defined by:

$$S(G_{T|C}; G_C; A) = \{V^{\sim}, V^{-}, E^{\sim}, E^{-}, E^{+}\}. \tag{6.1}$$

Note in particular that the matching is called *perfect* provided that the three sets V^{-}, E^{-}, E^{+} are empty, in which case $G_{T|C}$ is isomorphic to an induced sub-graph of G_C.

6.2.3 Assessing a Template in a Hasse Diagram

Perfect matching. Along the growth process, we are interested in the complexes C which exhibit a perfect matching with the associated restricted template $G_{T|C}$, and which are maximal—there exists no perfect matching for the successors of C. Such complexes are easily obtained from the matchings provided by a MCIS calculation between the graphs G_C and $G_{T|C}$ in each node of the Hasse diagram. Note that a perfect matching contains at most one pair of nodes (one from $G_{T|C}$, one from G_C) per protein type in C since $G_{T|C}$ has at most one node per protein type in C.

Alternate matching. Consider a complex C such that there is no perfect matching for C or any of its successors. In that case, we are interested in maximizing the number of common contacts between G_C and $G_{T|C}$, which corresponds to a MCES calculation. To report such matchings, we proceed as follows. First, for each complex C which is a root of the Hasse diagram, we compute the MCES between G_C and $G_{T|C}$. Second, let A be a matching returned by the MCES calculation. We search the ancestor D of C involving the protein instances and contacts of C matched by A, and minimizing the number of extra contacts. Note that, as for the perfect matching, an alternate matching contains at most one pair of nodes (one from $G_{T|C}$, one from G_C) per protein type in C.

6.3 Results: Y-complex Analysis

In this section, we present matching results, based on the templates presented in Chapter 4.

6.3.1 Perfect Matching

Perfect matchings reflect the largest sub-complexes of the Y-complex without any missing or extra contact w.r.t. the model—see the rows tagged with $G_t(Y)$ in the Table 6.1.

We first classify the matchings following their similarity with known sub-complexes of the Y-complex: 16 copies of the Y_X-tail ($M_P(1), M_P(2)$), 12 of the Y_X-edge (five split in two subunits ($M_P(3), M_P(4)$); seven entire units cf $M_P(5)$), 12 of the Y_X-short-arm ($M_P(6), M_P(7)$) and 16 of the Y_X-long-arm ($M_P(8)$). In addition, we have four perfect matchings corresponding to the Y-core ($M_P(9)$). The four remaining perfect matchings ($M_P(10)$) have one matching protein type i.e. Sec13. Let us inspect these perfect matchings.

The sixteen perfect matchings involving the Y_X-tail ($M_P(1), M_P(2)$) show that Nup133 and Nup84 are well positioned w.r.t. one another. But the absence of larger perfect matching shows that in the fourteen complexes under investigation in $M_P(1)$, Nup133 makes an erroneous contact with Nup145C.

The 12 perfect matchings involving parts of the Y_X-edge show that there is an additional contact between Sec13 and Nup84 in the model for at least five copies of the Y_X-edge ($M_P(3), M_P(4)$).

The contact between Y_X-short-arm and Nup85 appears in 16 perfect matchings ($M_P(6), M_P(7), M_P(9)$), while the one between Y_X-short-arm and Nup145C appears in only five matchings ($M_P(7), M_P(9)$). As illustrated by the Figure 5.11, each copy of the Y-complex are split into two pieces. The 16 perfect matchings of the Y_X-long-arm ($M_P(8)$) show a good relative position between Seh1 and Nup85. The same holds for the four protein of Y-core ($M_P(9)$), as evidenced by the four matchings.

Finally, the matchings involving Sec13 involve protein instances without any valid contact. Their positioning appears as uncertain, a fact likely related to the small size of Sec13. (With a molecular weight of 41 kDa, Sec13 is the smallest one of the NPC.) As a matter of fact, no available data for the position of Sec13 is found in [ADV$^+$07a, supplemental Table 7]. Interestingly, the volume ratios associated with these results are bounded by 2.57 ($M_P(9)$).

Template; tag	#	$V^{\sim}$	$\min \overline{V}_\lambda$	$\max \overline{V}_\lambda$
$G_t(Y)$;$M_P(1)$	14	Y_X-tail	0.77	0.90
$G_t(Y)$;$M_P(2)$	2	(Y_X-tail,Nup145C)	0.85	0.88
$G_t(Y)$;$M_P(3)$	5	(Nup145C,Nup84)	0.81	0.88
$G_t(Y)$;$M_P(4)$	5	(Nup145C,Sec13)	0.81	0.86
$G_t(Y)$;$M_P(5)$	7	Y_X-edge	0.78	0.88
$G_t(Y)$;$M_P(6)$	11	(Y_X-short-arm,Nup85)	0.88	0.91
$G_t(Y)$;$M_P(7)$	1	Y-junction	1.78	1.78
$G_t(Y)$;$M_P(8)$	16	(Y_X-long-arm)	0.77	1.63
$G_t(Y)$;$M_P(9)$	4	Y-core	1.15	2.57
$G_t(Y)$;$M_P(10)$	4	Sec13	0.58	0.69

Table 6.1: Perfect matchings for the templates $G_t(Y)$. Each matching is identified by a tag ($M_P(i)$) referenced in the text. The columns read as follows: $V^{\sim}$: protein types involved in the matching; #: number of identical matchings; $\min \overline{V}_\lambda(C)$ and $\max \overline{V}_\lambda(C)$: min and max volume ratios amidst identical matchings.

6.3.2 Alternate Matching

Alternate matchings aim at maximizing the number of common contacts, and involve largest sub-complexes of the Y-complex. We first computed matchings with $G_t(Y)$, see the Table 6.2. We get 11 for (Y_X-tail,Y_X-edge) ($M_A(1), M_A(2)$), and 11 for (Y_X-short-arm,Y_X-long-arm) ($M_A(3)$). These matchings correspond to eleven copies of the Y-complex split in two sub-complexes. Together with the five matchings for the whole Y-complex ($M_A(4), M_A(5)$), we get an overall stoichiometry of 16, as expected.

The number of extra contacts observed is bounded by seven for a maximum of fifteen. (Seven proteins make at most twenty one pairwise contacts, out of which six belong to the template.) Note that the only missing protein type in the five matchings involving Y-main is restricted to Sec13 in three of them. Note also that the volume ratio does not exceed 4.71 ($M_A(3)$).

Template; tag	#	$V^{\sim}$	$\mid V^{-}\mid$	$\mid E^{\sim}\mid$	$\mid E^{-}\mid$	min $\mid E^{+}\mid$	max $\mid E^{+}\mid$	$\min \overline{V}_{\lambda}$	$\max \overline{V}_{\lambda}$
$G_t(Y)$;$M_A(1)$	1	(Y_X-tail,Nup145C)	4	2	0	1	1	3.43	3.43
$G_t(Y)$;$M_A(2)$	10	(Y_X-tail,Y_X-edge)	3	3	0	3	3	3.61	4.32
$G_t(Y)$;$M_A(3)$	11	Y-arms	4	2	0	1	1	0.94	4.71
$G_t(Y)$;$M_A(4)$	3	(Y-main,Seh1)	1	5	0	2	6	1.13	2.96
$G_t(Y)$;$M_A(5)$	2	Y-complex	0	6	0	7	7	3.37	3.47

Table 6.2: Alternate matchings for the template $G_t(Y)$. Each matching is identified by a tag ($M_A(i)$) referenced in the text. The columns read as follows: #: number of identical matchings; $V^{\sim}$: protein types involved in the matching; $\mid V^{-}\mid, \mid E^{\sim}\mid, \mid E^{-}\mid, \min\mid E^{+}\mid, \max\mid E^{+}\mid$: size of the sets involved in the signature of the matching—min and max taken amidst all identical matchings; $\min \overline{V}_{\lambda}$ and $\max \overline{V}_{\lambda}$: min and max volume ratios amidst identical matchings.

6.3.3 Further In-silico Experiments

Removing Sec13 from the toleranced model. Due to the poor location of Sec13 in the assembly, we remove it from the toleranced model and compute the new matching. The perfect matchings of Table 6.3 do not reveal a significant improvement: the perfect matchings observed involve more protein types (e.g Y-core in $M_P(9)$), but require a larger volume ratio. The same holds for alternate matchings, see Table 6.4.

Template; tag	#	$V^{\sim}$	$\min r_{\lambda}$	$\max r_{\lambda}$
$G_t(Y)$;P_1	13	Y_X-tail	0.77	0.90
$G_t(Y)$;P_2	3	(Y_X-tail,Nup145C)	0.85	0.87
$G_t(Y)$;P_3,P_4,P_5	7	(Nup145C,Nup84)	0.81	0.88
$G_t(Y)$;P_6	9	(Y_X-short-arm,Nup85)	0.88	0.91
$G_t(Y)$;P_7	1	Y-junction	2.26	2.26
$G_t(Y)$;P_8	16	(Y_X-long-arm)	0.77	1.54
$G_t(Y)$;P_9	6	Y-core	1.10	3.05

Table 6.3: Perfect matchings of $G_t(Y)$. Sec13 was removed from the toleranced model. The tags $M_P(i)$ match those used in Table 6.1.

Template; tag	#	$V^{\sim}$	$\mid V^{-}\mid$	$\mid E^{\sim}\mid$	$\mid E^{-}\mid$	max $\mid E^{+}\mid$	min $\mid E^{+}\mid$	$\min r_{\lambda}$	$\max r_{\lambda}$
$G_t(Y)$;A_1,A_2	9	(Y_X-tail,Nup145C)	4	2	0	1	1	3.03	3.82
$G_t(Y)$;A_3	9	(Y-arms)	4	2	0	1	1	0.94	4.79
$G_t(Y)$;A_4,A_5	7	(Y-main,Seh1)	1	5	0	2	6	1.09	3.45

Table 6.4: Alternate matchings of $G_t(Y)$. Sec13 was removed from the toleranced model. The tags $M_A(i)$ match those used in Table 6.2.

6.4 Results: T-complex Analysis

As opposed to the Y-complex, no (sub-)complex of the T-complex has been crystallized. In the following, we therefore investigate the coherence between putative pairwise contacts in the T-complex, and the copies of the T-complex embedded in the toleranced model of the NPC.

6.4.1 Perfect Matching

As summarized in the Table 6.5, we computed the perfect matchings w.r.t. the skeleton graph $G_t(T)$ for all nodes of the Hasse diagram. One gets sixteen perfect matchings corresponding to the T-leg ($M_P(11), M_P(12)$), fourteen to the T-core ($M_P(13)$) and two to the entire T-complex ($M_P(14)$). A careful inspection shows that all contacts of $G_t(T)$ are found in all copies of the T-complex. The low number of perfect matchings (2) owes to extra contacts, namely (Nup49 and Nsp1) and/or (Nup49 and Nic96) and/or (Nup57 and Nic96). Yet, we found Nup57 and Nic96 in 16 different perfect matchings ($M_P(13), M_P(14)$), showing that these two types do not make any contact.

Template; tag	#	$V^{\sim}$	min $\overline{V}_\lambda$	max $\overline{V}_\lambda$
$G_t(T)$;$M_P(11)$	10	T-leg	0.57	0.75
$G_t(T)$;$M_P(12)$	6	(T-leg,Nsp1)	0.61	0.72
$G_t(T)$;$M_P(13)$	14	(T-core,Nup57)	0.74	1.37
$G_t(T)$;$M_P(14)$	2	T-complex	1.79	2.28
$G_t(T\text{-comp})$;$M_P(15)$	2	T-leg	2.42	2.79
$G_t(T\text{-comp})$;$M_P(16)$	16	(T-leg,Nsp1)	0.61	0.81
$G_t(T\text{-comp})$;$M_P(17)$	5	T-core	0.79	1.46
$G_t(T\text{-comp})$;$M_P(18)$	10	(T-core,Nup49)	0.97	1.76
$G_t(T\text{-comp})$;$M_P(19)$	1	(T-core,Nup57)	1.91	1.91
$G_t(T\text{-new})$;$M_P(20)$	6	(T-leg,Nsp1)	0.61	0.81
$G_t(T\text{-new})$;$M_P(21)$	2	(T-leg,Nic96)	2.13	2.25
$G_t(T\text{-new})$;$M_P(22)$	6	(T-core,Nup57)	0.78	1.37
$G_t(T\text{-new})$;$M_P(23)$	10	T-complex	0.98	1.73

Table 6.5: Perfect matchings for the templates $G_t(T)$, $G_t(T\text{-comp})$ and $G_t(T\text{-new})$. Each matching is identified by a tag ($M_P(i)$) referenced in the text. The columns read as follows: $V^{\sim}$: protein types involved in the matching; #: number of identical matchings; min $\overline{V}_\lambda(C)$ and max $\overline{V}_\lambda(C)$: min and max volume ratios amidst identical matchings.

6.4.2 Alternate Matching

As seen from the Table 6.6, 18 alternate matchings of the entire T-complex are found ($M_A(6)$), for a volume ratio less than 2.36. Further investigation shows two instances of Nup49, each interacting with two instances of Nup57 belonging to two copies of the T-complex, contribute to two extra matchings—whence 18 matchings and not 16. These spurious matching tough, are easily ruled out from the λ value for which contacts between Nup49 and Nup57 appear, since the second contact appears at $\lambda = 0.38$, after the last merge of complexes at $\lambda = 0.33$. The analysis of alternate matchings also exhibits at most two extra contacts between Nup49 and Nsp1, and between Nup49 and Nic96.

Template; tag	#	$V^{\sim}$	$\mid V^{-} \mid$	$\mid E^{\sim} \mid$	$\mid E^{-} \mid$	min $\mid E^{+} \mid$	max $\mid E^{+} \mid$	$\min \overline{V}_{\lambda}$	$\max \overline{V}_{\lambda}$
$G_t(T)$;$M_A(6)$	18	T-complex	0	3	0	0	2	0.83	2.36
$G_t(T$-comp$)$;$M_A(7)$	11	T-complex	0	5	1	0	0	0.98	1.85
$G_t(T$-comp$)$;$M_A(8)$	7	T-complex	0	4	2	0	0	1.17	2.22
$G_t(T$-comp$)$;$M_A(9)$	4	T-complex	0	3	3	0	0	1.80	2.28
$G_t(T$-new$)$;$M_A(10)$	10	T-complex	0	5	0	0	0	0.98	1.73
$G_t(T$-new$)$;$M_A(11)$	8	T-complex	0	4	1	0	1	1.17	2.26
$G_t(T$-new$)$;$M_A(12)$	4	T-complex	0	3	2	0	0	1.79	2.29

Table 6.6: Alternate matchings for the templates $G_t(T), G_t(T\text{-comp})$ and $G_t(T\text{-new})$. Each matching is identified by a tag ($M_A(i)$) referenced in the text. The columns read as follows: #: number of identical matchings; $V^{\sim}$: protein types involved in the matching; $\mid V^{-} \mid, \mid E^{\sim} \mid, \mid E^{-} \mid, \min \mid E^{+} \mid, \max \mid E^{+} \mid$: size of the sets involved in the signature of the matching—min and max taken amidst all identical matchings; $\min \overline{V}_{\lambda}$ and $\max \overline{V}_{\lambda}$: min and max volume ratios amidst identical matchings.

6.4.3 Further In-Silico Experiments

Testing new templates of the T-complex. To single out frequent contacts not present in $G_t(T)$, we consider the complete skeleton graph $G_t(T\text{-comp})$. We obtain 18 perfect matchings corresponding to the T-leg ($M_P(15), M_P(16)$) and 16 to the T-core ($M_P(17), M_P(18), M_P(19)$). These matchings highlight two relevant contacts absent from the skeleton $G_t(T)$: Nup49 and Nsp1 obtained 16 times ($M_P(16)$), and Nup49 and Nic96 obtained ten times ($M_P(18)$). Adding these contacts to the skeleton graph $G_t(T)$ yields $G_t(T\text{-new})$. This new graph yields eight perfect matchings with T-leg ($M_P(20), M_P(21)$), six to the T-core ($M_P(22), M_P(23)$) and ten to the entire T-complex ($M_P(23)$). We note that the number of perfect matchings with the entire T-complex node set moves from two for $G_t(T)$ to ten for $G_t(T\text{-new})$.

In terms of alternate matchings, $G_t(T\text{-new})$ yields 22 alternate matchings, containing in particular the ten perfect matchings already discussed—the matchings counted in the lines $M_P(23)$ and $M_A(10)$ are in one-to-one correspondence. The extra 12 matchings owe again to contacts between protein instances of different copies of the T-complex. These extra matchings involve at most two missing contacts corresponding to the contacts added w.r.t. $G_t(T)$. Also, these matchings do not have any extra contact, except one corresponding to the a contact between Nup57 and Nic96, see line $M_A(11)$.

6.5 Artwork

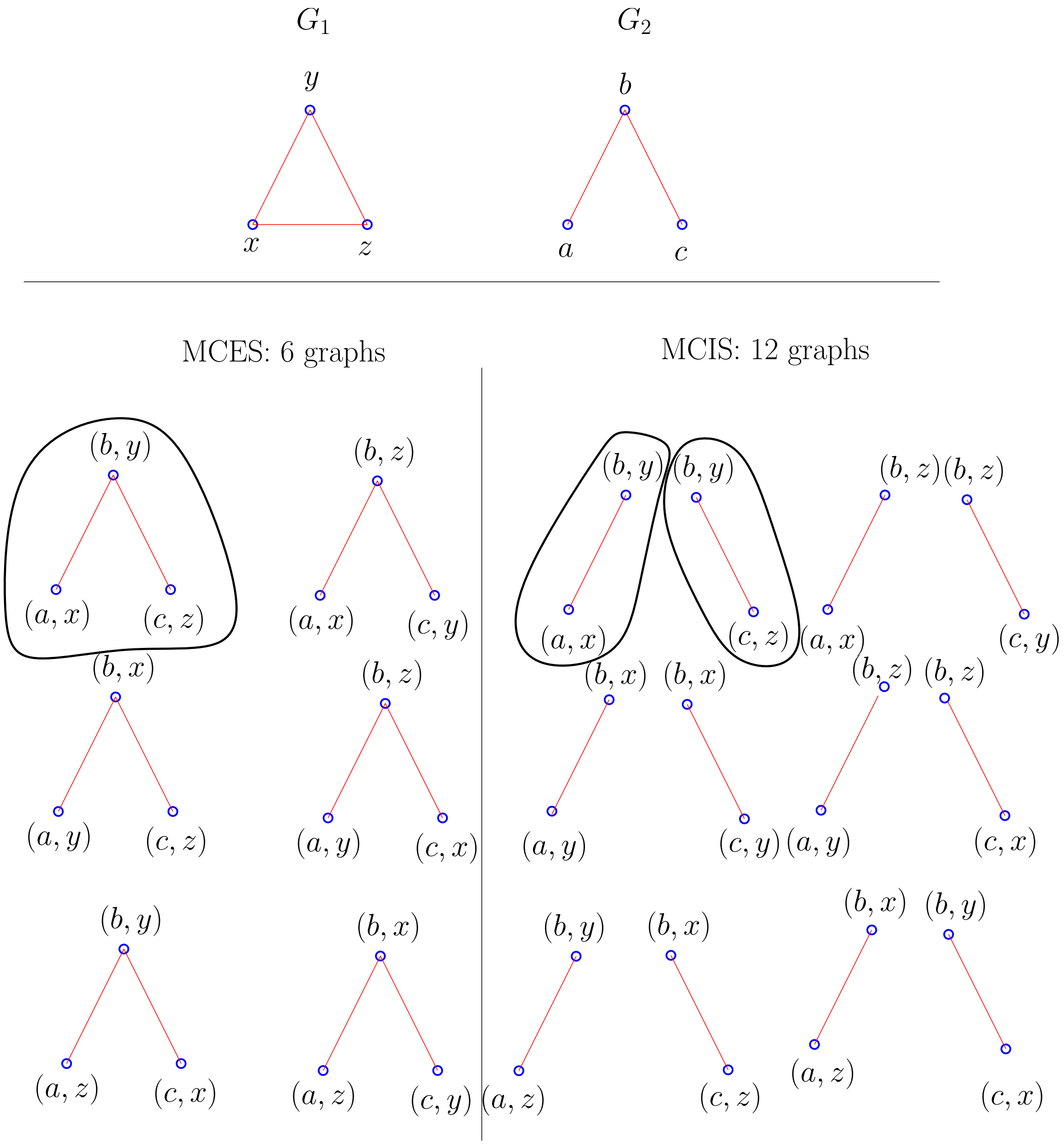
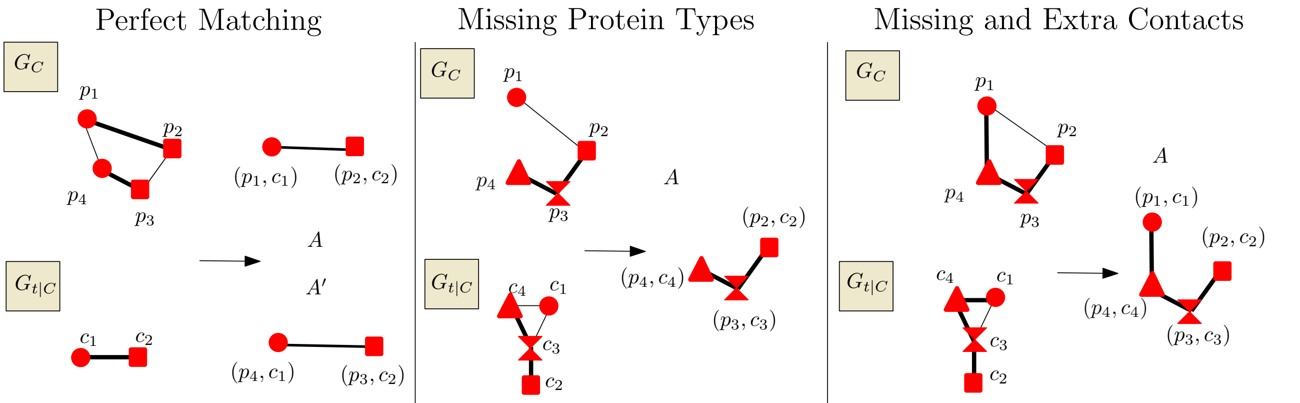


Figure 6.1: Comparing graphs with matchings: illustration of the Maximal Common Edge Sub-graph (MCES) and Maximal Common Induced Sub-graph (MCIS) constructions. **Top.** Two labelled graphs G_1 and G_2. **Bottom Left.** The 6 MCES of G_1 and G_2 **Bottom Right.**: The 12 MCIS of G_1 and G_2. If we impose a correspondence between labels $((a,x),(b,y),(c,z))$, there is one MCES and there are two MCIS, namely the circled graphs.


Perfect Matching

Missing Protein Types

Missing and Extra Contacts

Matchings	A	A'
Matching Protein Types	(p_1, c_1) (p_2, c_2)	(p_3, c_2) (p_4, c_1)
Matching Contacts	(c_1, c_2) $\leftrightarrow$ (p_1, p_2)	(c_1, c_2) $\leftrightarrow$ (p_4, p_3)
Missing Protein Types	$\emptyset$	$\emptyset$
Missing Contacts	$\emptyset$	$\emptyset$
Extra Contacts	$\emptyset$	$\emptyset$

Matchings	A
Matching Protein Types	(p_2, c_2) (p_3, c_3) (p_4, c_4)
Matching Contacts	$(c_2, c_3) \leftrightarrow (p_2, p_3)$ $(c_4, c_3) \leftrightarrow (p_4, p_3)$
Missing Protein Types	c_1
Missing Contacts	$\emptyset$
Extra Contacts	$\emptyset$

Matchings	A
Matching Protein Types	(p_1, c_1) (p_2, c_2) (p_3, c_3) (p_4, c_4)
Matching Contacts	$(c_2, c_3) \leftrightarrow (p_2, p_3)$ $(c_4, c_3) \leftrightarrow (p_4, p_3)$ $(c_4, c_1) \leftrightarrow (p_4, p_1)$
Missing Protein Types	$\emptyset$
Missing Contacts	(c_1, c_3)
Extra Contacts	(p_1, p_2)

Figure 6.2: Signature of a matching between the skeleton graphs G_C of a complex C and $G_{T|C}$ of a template T restricted to C. The match between a protein instance of G_C and a type of $G_{T|C}$ is materialized by an identical geometric shape (disk, square, triangle, hourglass). Matched contacts corresponds to bold edges. The adjectives matching/missing/extra qualify G_C w.r.t. $G_{T|C}$. **(Left.)** A Maximal Common Induced Sub-graph calculation yields two perfect matchings. **(Middle.)** A Maximal Common Edge Sub-graph calculation yields a matching with one missing protein type. **(Right.)** A Maximal Common Edge Sub-graph calculation yields a matching with missing and extra contacts.

6.6 Supplemental

6.6.1 Algorithms

The calculation of all maximal common sub-graphs of two graphs G_1 and G_2 is equivalent to the enumeration of all maximal cliques of a so-called product graph [Koc01], a problem for which exact algorithms were proposed in [CK05]. In fact, there are two kind of product graphs:

- the *edge product graph*, from which we generate Maximal Common Edge Sub-graphs (MCES). Each node of the *edge product graph* is associated to a pair of edges $(e_1 \in E[G_1], e_2 \in E[G_2])$, and there is an edge between two nodes (e_1, e_2) and (f_1, f_2) iff (e_1, f_1) and (e_2, f_2) are incident together to a same vertex, or are not incident together to a same vertex.

- the *vertex product graph*, from which we generate Maximal Common Induced Sub-graphs (MCIS). Each node of the *vertex product graph* is associated to a pair of vertices $(u_1 \in V[G_1], u_2 \in V[G_2])$, and there is an edge between two nodes (u_1, u_2) and (v_1, v_2) iff (u_1, v_1) and (u_2, v_2) are neighbors together, or are not neighbors together.

Note that the definition of product graphs is purely topological. But in our setting, a protein type is associated to each vertex of graphs G_1 and G_2, and we only match two vertices provided that they carry the same protein type. Similarly, matching two edges requires the agreement of their vertices. As an example, consider Figure 6.3 and assume that a matches x, that b matches y and that c matches z. Under these hypothesis, there is a single MCES and two MCIS.

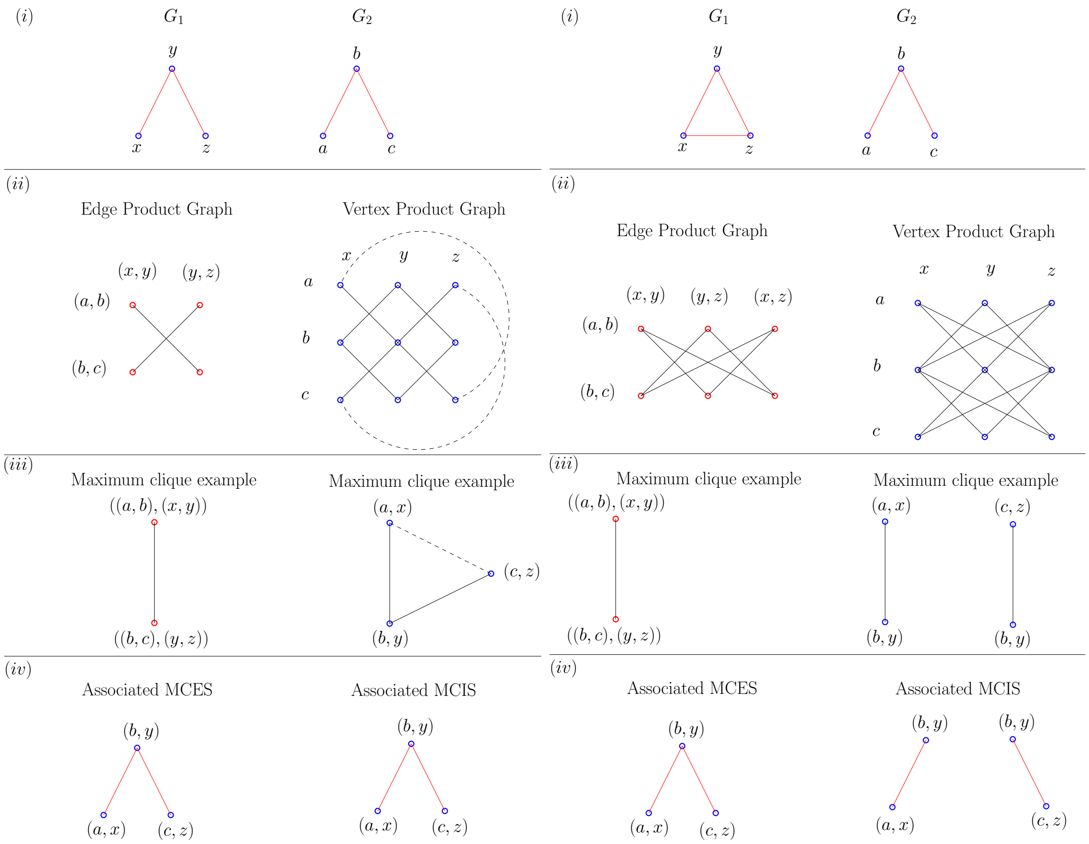


Figure 6.3: Two illustrations of MCS resolution via Maximum clique problem. **(i).** Input are two labelled graphs G_1 and G_2. Vertices are drawn in blue and edges in red. **(ii).** Product Graphs of G_1 and G_2. Nodes of Edge (resp. Vertex) Product Graph represent pair of edges (resp. vertices) of G_1 and G_2. Two nodes in the product graph are linked by a solid edge if they are incident in G_1 and G_2. Two nodes in the product graph are linked by a dashed edge if they are not incident in G_1 and G_2. **(iii).** Examples of maximum cliques from the product graphs. **Left example**: There are 2 maximum cliques for the Edge Product Graph and 2 maximum cliques for the Vertex Product Graph. **Right example**: There are 6 maximum cliques for the Edge Product Graph and 12 maximum cliques for the Vertex Product Graph.**(iv).** Conversion from Product Graphs to MCS of G_1 and G_2.

Chapter 7

Software

7.1 Introduction - Rationale

The VORATOMsoftware suite, for Voronoi Analysis of Toleranced Models, is a set of tools meant to design and analyze toleranced models of macro-molecular assemblies. The suite consists of the programs presented on Figure 7.1, which are all encapsulated within on main executable, also called VORATOM. These executables are presented in section 7.2, while section 7.3 comments on the underlying C++ design. All can be downloaded from `http://cgal.inria.fr/abs/voratom`.

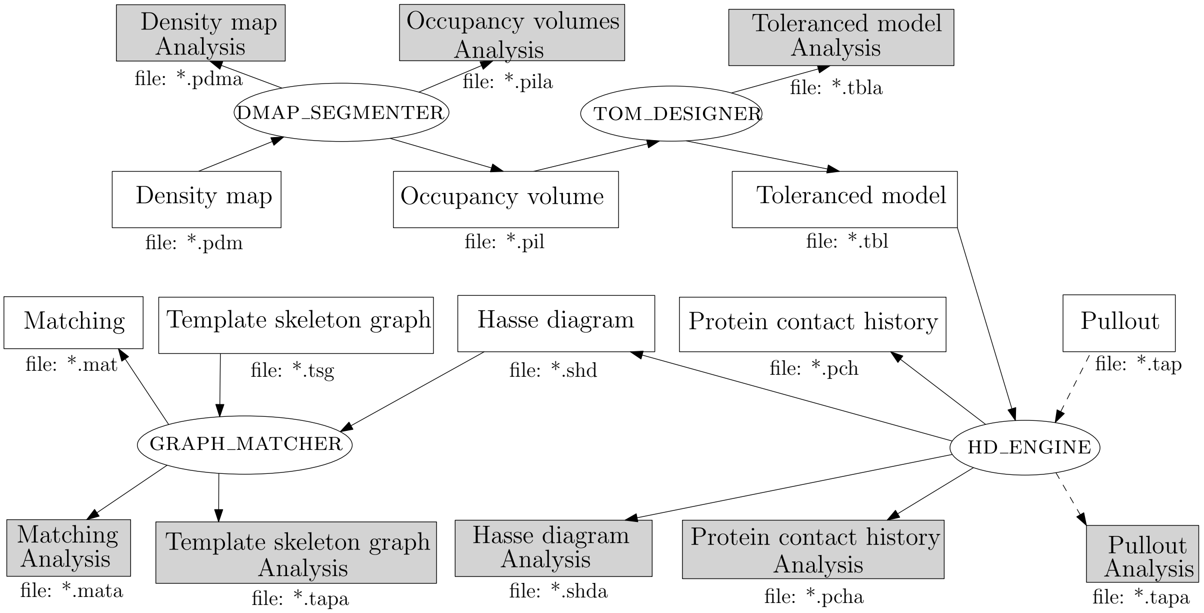


Figure 7.1: Overview of the applications: ellipsis represent executable programs; unfilled rectangles represent the main objects; gray rectangles represent the analysis associated to instances of the main objects.

7.2 Overview: Applications and File Formats

7.2.1 Overall application

Given a set of maps, the VORATOM building blocks, which are also made available on a stand-alone basis, perform the segmentation of the maps into occupancy volumes (section 7.2.2), create a toleranced model from these occupancy volumes (section 7.2.3), and compute the Hasse diagram associated to the toleranced model (section 7.2.4).

In presenting the executables, we also report the running times obtained on a dual core Intel Extreme CPU X7900 2.80GHz with RAM size of 8Go, running Fedora Core 14. We also note that our programs, written in C++, were compiled with g++ at the optimization level -O3.

- DMAP_SEGMENTER
  The computation of the occupancy volumes of all the 33 maps was done in 28.3 seconds (18 seconds for loading all the maps and 10.3 seconds for the computations). A total of 448 protein instances were reported.

- TOM_DESIGNER
  The computation of the toleranced model of the NPC from the 448 occupancy volumes was done in less than one second. A total of 8064 toleranced balls were reported.

- HD_ENGINE
  The computation of the protein contact history from the 8064 toleranced balls with $\lambda_{\max} = 1$ failed. We used the partial λ-complex that ran in 72.5 seconds. 2507 protein contacts were reported. The computation of the Hasse diagram from this protein contact history was done in 9.9 seconds.

- GRAPH_MATCHER
  The computations of all perfect and alternate matching of a template skeleton graph with complexes in a Hasse diagram required less than one second. For $G_t(Y)$ (respectively $G_t(T)$, $G_t(T\text{-comp})$ and $G_t(T\text{-new})$), a total of 69 (32, 34 and 24) perfect matching and 27 (18, 22 and 22) alternate matching were reported.

7.2.2 Density map segmenter

Consider a map, namely a 3D matrix with one number $\in [0,1]$ per voxel.
The map segmenter, named DMAP_SEGMENTERin the sequel, selects from the map a prescribed set of connected regions called occupancy volumes. The algorithm is described in Section 4.4.

Input. The main argument is a map. See the .pdm file format on Figure 7.2.

Output. The main output is a list of occupancy volumes, one per protein instance. A given occupancy volume is represented by the (x, y, z) coordinates of the voxels allocated to this instance, and the density of each voxel is also reported. See the .ovl file format on Figure 7.3.

Analysis. There are two analysis. The first one, at the protein instance level, compares the occupancy volume against the protein reference volume (the volume estimated from its sequence). The second one, at the map level, compares the number of occupancy volumes created versus the stoichiometry of the protein. (Typically, if the stoichiometry of the protein is larger than the number of connected components of voxels with a non null probability, it may not be possible to create the desired number of instances.)

```
# Global attributes of the map: name of the protein type,
# stoichiometry, number nv of voxels along the x y and z directions
Nup84 16 100
# Cartesian coordinates of the bottom-left corner
x y z

# Densities: nv * nv * nv real numbers, each in the range 0..1
0 0 0 0...
```

Figure 7.2: The .pdm file format to represent a cubic map—the number of voxels is the same along each direction.

```
# Global attribute: total number of occupancy volumes
448

# Then, we find the list of occupancy volumes. Here is one example,
# namely an instance of the protein type Nup192:
# Nup192: protein type;  0: instance index; 240: number of voxels of
# the occupancy volume attributes to this instance
Nup192 0 240
# Then, a list of 240 voxels; for each, the  Cartesian coordinates
# xyz,  and the probability density value
45 41 17 1
...
```

Figure 7.3: The .ovl file format to represent the occupancy volumes of a list of protein instances.

7.2.3 Toleranced Model Designer

The Toleranced Model Designer, called TOM_DESIGNERin the sequel, computes a toleranced model from a list of occupancy volumes. Note that each toleranced protein consists of a list of toleranced balls. The algorithm used to compute the canonical shapes of the proteins is presented in Section 4.4.2.

Input. The main argument is a list of occupancy volumes, each corresponding to one protein instance. (See the .ovl file format.)

Output. The output is the toleranced model. See the .tbl file format on Figure 7.4.

Analysis. Given a λ value and a list of protein instances, the analysis of a toleranced model consists of computing the volume ratios of Eq. (5.1). Note that the volumes of these instances are computed amidst the whole NPC. Also, the volume calculation is carried out using affine α-shapes [CKL11], as computing the volume of restrictions in the compoundly weighted Voronoi diagram is an open problem.

```
# Global attribute: total number of toleranced balls
8064

# Then, a list of toleranced balls. Here is an example toleranced
# ball, represented by the Cartesian coordinates of the center, the
# inner radius, the outer radius, and the index of the protein instance
# this ball belongs to
42.4916 39.2358 16.4461 1.4702 4.19091 0
...
```

Figure 7.4: The .tbl file format to represent the toleranced balls of a list of protein instances.

7.2.4 Hasse Diagram Engine

The Hasse Diagram engine, called HD_ENGINEin the sequel, computes the Hasse diagram of a toleranced model, as explained in Section 5.2.2

This computation requires two steps, namely the computation of the λ-complex of the toleranced model, and that of the Hasse diagram of the protein complexes. While computing the Hasse diagram, we also store the list of merges between pairs of protein instances, which we call the *protein contact history*. In fact, we successively compute (i) the λ-complex, (ii) the protein contact history, (iii) the Hasse diagram.

The file formats for the Hasse diagram and the protein contact history are presented on Figures. 7.6 and 7.7 .

Input. While the main argument is the toleranced model, the following options are available:

- One can specify a list of pulldowns; if so, the toleranced model manipulated falls into the bicolor setting.
- If a protein contact history is provided, the Hasse diagram is directly derived from it, without computing the λ complex.
- A value $\lambda_{\max}$ can be specified to bound the growth process of the toleranced model. Recall that this value should be set in accordance with the uncertainties observed on the input data, measured by volume ratios.
- Since the computation of the whole λ complex may be time consuming, the partial λ-complex, which consists of the Gabriel simplices of dimension zero and one, may be resorted to. One option is provided to resort to this subset of the λ-complex [CD10].

Output. If no protein contact history is provided, the one computed from the (partial) λ-complex one is reported. In any case, we report the Hasse diagram involving all protein instances. Furthermore, one finds one Hasse diagram per pulldown specified, if any.

Analysis. The following pieces of information are reported:

- The lifetime of the complexes found in the Hasse diagram, and the number of complexes as a function of λ.
- The isolated copies found in the Hasse diagram. (Recall that a pulldown is mandatory to define isolated copies.)
- The contact probabilities of protein type pairs, as defined in Section 5.2.2. Note that the contact probabilities requires a value for $\lambda_{\max}$. If such a value has not been specified, $\lambda_{\max} = 1$ is used.
- The volume ratio of each complex found in the Hasse diagram is computed, for the λ corresponding to its birth date.

```
# An example pulldown.
# A pulldown is represented by a triple namely
#(i)   pulldown index
#(ii)  the number of protein types in the pulldown
#(iii) the list of protein types, the first one being the tagged
#      protein
#
# As an example, here is the pulldown of the Y-complex in Sali et al:
54 7 Nup84 Seh1 Nup85 Nup120 Nup145C Sec13 Nup133
...
```

Figure 7.5: The .tap file format to represent a pulldown i.e. a list of protein types.

```
# An example protein contact history.
# An element is represented by a triple namely
#(i)   protein instance p1 (type + index)
#(ii)  protein instance p2 (type + index)
#(iii) the weight for which p1 and p2 are connected.
Nup84 55 Nup133 89 0.515
...
```

Figure 7.6: The .pch file format to represent a protein contact history.

```
# Global attribute: pulldown, see .tap file format
54 7 Nup84 Seh1 Nup85 Nup120 Nup145C Sec13 Nup133
# total number of vertices and edges in the diagram
495 493

# Vertex description: vertex index and weight of the vertex
66 -0.289302
# Then, description of the protein complex of the vertex:
# number of vertices and edges in
2 1
# list of vertices of the protein complex: pair (type name,
# instance index) of the protein instance in the vertex
Nup133 73
Nup84 114
# list of edges of C: two pairs (type name, instance index) following
by the weight of the edge linking the instances
Nup133 73 Nup84 114 -0.289302

# Edge description: vertex indices of the ancester and the son
66 52
```

Figure 7.7: The .shd file format to represent a Hasse diagram. In the example, the protein complex of the vertex number 52 (not shown) has one protein instance (Nup84, 114).

7.2.5 Graph Matcher

This application compares the skeleton graphs of nodes of a Hasse diagram to a template graph involving the same protein types, typically corresponding to an atomic resolution model of the corresponding sub-complex—see Section 6.2.3.

Input. The two main arguments are a (list of) Hasse diagrams (.shd file format) and a (list of) template skeleton graph(s) (.tsg file format). A .tsg file describes a graph whose nodes are protein types (Figure 7.8).

Output. The main output is a list of (alternate and perfect) matching recorded in a .mat file (Figure 7.9).

Analysis. The analysis on perfect and alternate matchings essentially consists of computing their signature as defined in Section 6.2.2. A table summarizing the signatures is dumped into the analysis file (.mata).

```
# Global attribute: number of edges
6

# list of edges
Nup133 Nup84
...
```

Figure 7.8: The .tsg file format to represent a template skeleton graph.

```
#Global attribute: template skeleton graph
1
Nup133 Nup84

#Global attribute: protein complex graph
2 1
Nup133 73
Nup84 114
Nup133 73 Nup84 114 -0.289302

#Global attribute: matching type (perfect or alternate)
Alternate Matching


#Global attribute: number of matching of this type
1

#list of all matching of this type: number of edges following by the
list of edges
1
Nup133 73 Nup84 114 -0.289302
```

Figure 7.9: The .mat file format to represent a matching between a template skeleton graph and a protein complex graph.

7.3 Design of the Packages

7.3.1 Overview

Packages. The code has been written in generic C++, in the spirit of the Computational Geometry Algorithms Library, see `http://www.cgal.org`. A particular care has been taken in separating the numerical operations (the so-called predicates and constructions) from the combinatorial ones.
Each of the aforementioned application corresponds to one *package*, that is to a set of C++ classes addressing a specific problem. We have developed the following packages, each one with its own C++ namespace:

- package *Geometry*;
- package *Graph_theory*;
- package *Biochemical*;
- package *Density_map_segmenter*;
- package *Toleranced_model_designer*;
- package *Hasse_diagram_engine*;
- package *Graph_matcher*.

Type of classes within a package. The classes within a package are divided into three groups, as seen from Figure 7.10:

- A *kernel* class contains constant size objects, predicates and constructions. Constant size objects are C++ classes providing representations and basic operations (comparison, input from a stream, output into a stream), and are grouped within the *Kernel_base* class. Predicates, grouped in the *Kernel_predicates* class, are C++ structures providing functors i.e. function objects answering a query and returning an output within a finite set, typically true or false. Constructions are C++ structures providing functors building constant size objects, and are grouped in the *Kernel_constructions* class.
- A *data structure* class defines a combinatorial object used in the package. The specification of a data structure typically requires a kernel defining the objects of constant size stored, and a container storing these objects. Following the spirit of the Standard Template Library, the data structures have particular methods for filling or visiting them.
- An *algorithm* class implements a particular task, and directly depends upon the kernel for the geometric objects manipulated—and the related predicates and constructions. In practice, an algorithm is represented by a C++ class providing one ore more functor(s) whose arguments are inputs and outputs of the algorithm. When the output is a data structure, it is represented by a special iterator from which the data structure can be filled, making the algorithm independent from the data structure. When the input is a set of objects, it is represented by an iterator visiting this set.

Each class in a package is implemented so as to be generic, i.e to support different kinds of kernels, data structures or algorithms. We emphasize the fact that genericity provides flexibility, in the sense where the packages can be used in different contexts, for example with or without exact calculation, with different input format, or with different algorithms.

Categories of classes within a package. The classes of a package are grouped into categories, each defining types corresponding to a particular task:

- the Input category groups the input data structures of the algorithms;
- the Output category groups the output data structures of the algorithms;
- the Engine category groups the algorithms creating the output from the input;

- the Analysis category groups the algorithms analyzing the output computed by the algorithms.

Practically, a category is a C++ class encapsulating a number of C++ `typedef` instructions. That is, a category corresponds to a so-called C++ traits class. Note again that this design is meant to make the software suite very generic by abstracting and decoupling the types.

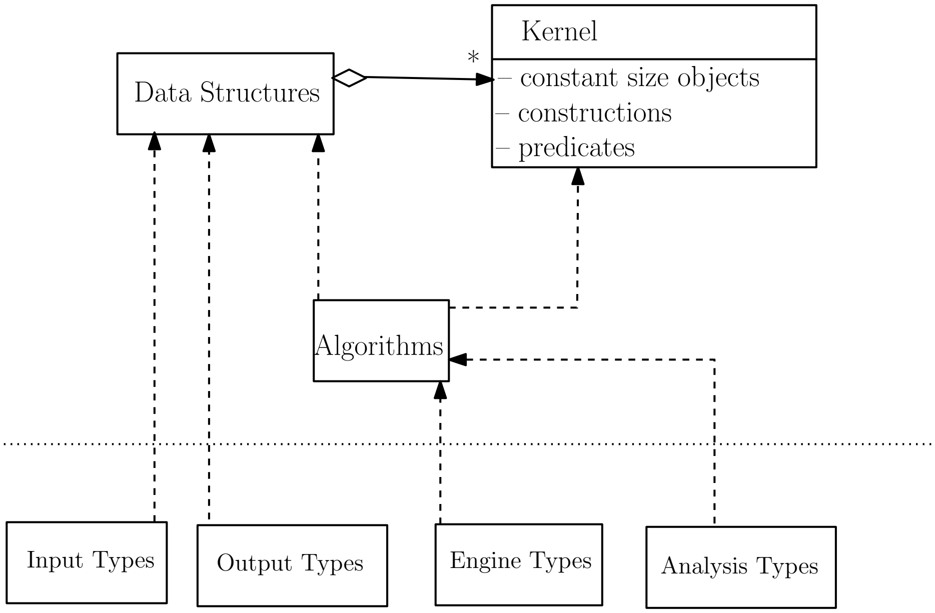


Figure 7.10: Dependence diagram of the C++ classes involved in a package, depicted using the UML formalism. A dashed arrow refers to a dependency relationship, while a solid arrow with a white diamond stands for the aggregation relationship. In the latter case, the * denotes that the concept on the diamond side contains many instances of the pointed concept. The dotted lines separate the types of classes and the categories of classes.

7.4 Packages: Details

In the following, for each package, we describe the main classes corresponding to the *kernel, data structure, algorithm* types mentioned above.

7.4.1 Geometry

This package provides generic C++ classes for representing and computing the compoundly weighted Voronoi diagram and its associated λ-complex. The reader is referred to Section 3.4.6 for a more detailed presentation.

Kernel. It defines types for:
– the numbers (coordinates, radii of spheres);
– the algebraic numbers (roots of degree four polynomials);
– the toleranced balls;
– the simplices of the dual structure.
Note that numbers are predefined in a geometric kernel of the CGAL library, and algebraic numbers are predefined in an algebraic kernel of the CGAL library – see Chapter II and IV in `http://www.cgal.org/Manual/latest/doc_html/cgal_manual/contents.html` for more details. In addition, the kernel defines three important predicates and constructions:
– the conflict-free predicate;
– the toleranced tangent predicate;
– the toleranced tangent balls construction.

Data structures. The data structures are:
– the compoundly weighted dual structure, that is a Hasse diagram of simplices representing the dual of a CW VD, see Section 3.3.2;
– the λ-complex, as defined in Section 3.3.4.

Algorithms. There are three algorithms: one computing the dual structure of the compoundly weighted Voronoi diagram, one for computing its associated λ-complex, and one computing directly the partial λ-complex.

7.4.2 Graph Theory

This package provides generic C++ classes for computing the Maximal Common Sub-graphs of two graphs, see Section 6.6.1. Since constant size objects, predicates and constructions are provided by external packages, this package provides only data structures and algorithms.

Data structures. The data structures are:
– a vertex labeled graph, represented by an adjacency graph from the Boost Graph Library, see `http://www.boost.org/doc/libs/1_47_0/libs/graph/doc/index.html`.
– a vertex product graph, that is a graph whose nodes are pairs of nodes of two vertex labeled graphs G_1, G_2, and edges are tagged by a binary value (c or d, see [CK05]).
– a edge product graph, that is a graph whose nodes are pairs of edges of two vertex labeled graphs G_1, G_2, and edges are tagged by a binary value (c or d, see [CK05]).

Algorithms. There are three algorithms: one computing the so-called maximal c-cliques of a graph whose edges are tagged by a binary value, one computing the Maximal Common Edge Sub-graph of two vertex labeled graphs, and one computing the Maximal Common Induced Sub-graph of two vertex labeled graphs [CK05].

7.4.3 Biochemical

This package provides tools for representing and analyzing biochemical data of the VORATOM suite.

Kernel. It defines types for:
– the protein types;
– the protein instances;
– numbers (coordinates, density values, etc);
– the voxel of a density map.

Data structures. The data structures are containers for the previous types:
– the TAP pulldown, that is an ordered set of protein types;
– the template skeleton graph, that is a graph whose nodes are protein types;
– the protein assembly, that is a list of protein instances;
– the probability density maps, that is a 3D matrix of voxels represented by a vector.

Algorithms. There is only one algorithm analyzing the probability density maps: the analysis computes the number of local maxima, the number of connected components of non null voxels, and compares the volume of each connected component to the reference volume Vol_{ref}.

7.4.4 Density Map Segmenter

This package provides algorithms segmenting a probability density map of a protein type P in n connected components of size at most $Vol_{ref}(P)$. In particular, it implements the two algorithms described in Section 4.4.

Kernel. The constant size objects defined in this kernel are identical to those of the biochemical kernel.

Data structures. The main data structure is the occupancy volume, that is a list of connected voxels with an attached protein instance.

Algorithms. There are three algorithms, two for computing the occupancy volumes from a probability density map, and one for analyzing the output occupancy volumes. The analysis computes from an occupancy volume *ov* the ratio between *ov* and the reference volume of its protein instance. If a list of occupancy volumes of the same protein type is provided, it also computes basic statistics over the volume ratio (minimum, mean and maximum)

7.4.5 Toleranced Model Designer

This package implements the construction of the toleranced model described in Section 4.4.2.

Kernel. The constant size objects defined in this kernel are identical to those of the biochemical kernel. It also defines the toleranced ball (see Section 3.2).

Data structures. The two main data structures are the toleranced protein, that is a list of toleranced balls with an attached protein instance, and the toleranced model, that is a list of toleranced proteins.

Algorithms. There are two algorithms. The first one computes a toleranced protein from an occupancy volume of a protein instance. The second one analyzes a set of toleranced proteins: given a λ value and a set of toleranced proteins C, the analysis computes for each protein the volume ratio $\overline{V}_{\lambda}(C)$ defined in Section 5.2.2. Note that this computation requires computation of the volume of a union of balls, which is done using the Vorlume software, see `http://cgal.inria.fr/abs/Vorlume/`, based upon the algorithm described in [CKL11].

7.4.6 Hasse Diagram Engine

This package implements the construction and the analysis of the Hasse diagram described in Section 5.2.

Kernel. The constant size objects defined in this kernel are identical to those of the toleranced model designer kernel.

Data structures. There are three main data structures:
– the protein contact history, that is an ordered set of triples (protein instance, protein instance, λ-value);
– the protein complex, that is a connected weighted edge graph whose nodes are protein instances, the weight of an edge being the λ-value corresponding to the contact between two proteins. – the Hasse diagram, that is a directed acyclic graph whose nodes are protein complexes.

Algorithms. There are six algorithms: three for computing the Hasse diagram associated to a toleranced model, and three for analyzing the output. The former three compute:
– the protein contact history of a toleranced model from its (partial) λ-complex;
– the filtered protein contact history that is the subset of an input protein contact history containing only triples with protein instances of types in a prescribed set T;
– the Hasse diagram associated to a protein contact history.
The latter three:
– compute the contact probabilities of all pairs of represented protein types, see Section 5.2.2, based on the protein contact history;
– report the number of protein complexes as a function of the parameter λ, see Section 5.2.2, based on the Hasse diagram;
– count the number of isolated copies, see Section 5.2.2, based on the Hasse diagram.

7.4.7 Graph Matcher

This package implements the construction and analysis of the matchings described in Section 6.2.3.

Kernel. The constant size objects defined in this kernel are identical to those of the biochemical kernel.

Data structures. The two main data structures are the template skeleton graphs and the protein complexes.

Algorithms. There are three algorithms: two for computing the alternate and perfect matchings of a protein complex and a template skeleton graph, and one for analyzing the output matchings. The analysis constructs the signature of all the matchings.

Chapter 8

Conclusion

This thesis makes three contributions, namely the study and the computation of compoundly weighted Voronoi diagram (Chapter 3), the assessment of ambiguous macro-molecular models (Chapter 4 and 5), and the comparison of protein interactions graphs in the context of macro-molecular assemblies (Chapter 6).

From an algorithmic standpoint (Chapter 3), we work out selected properties of the so-called compoundly weighted Voronoi diagram, a curved Voronoi diagram for which little was previously known. The fact that the bisectors are degree four algebraic surfaces creates topological complications more pronounced than those faced for other curved diagrams, such as the Apollonius or the Möbius diagram. Also, using a representation of the dual as a directed acyclic graph whose nodes are abstract (i.e. non embedded) simplices, we design a naive yet non trivial algorithm for the construction of such a diagram. Finally, we generalize the α-complex for the case of the compoundly weighted Voronoi diagram, a construction which we call the λ-complex.

As far as the assessment of macro-molecular assemblies is concerned, we provide novel methods for the analysis of ambiguous assemblies (Chapters 4 and 5). We introduce the notion of toleranced model, which allows representing shapes with uncertain contours with a continuum of models. We show that a multi-scale analysis of a toleranced model can be encapsulated in a Hasse diagram summarizing the evolution of contacts between proteins, from which global and local assessments can be inferred. At the assembly level, we introduce contact probability curves, which provide a stoichiometry dependent notion of contact between protein species. At the local level, we introduce a number of statistics summarizing the properties of protein complexes appearing in the Hasse diagram.
Applying these tools to the Nuclear Pore Complex, we confirm the disputed ring model of the Y-complex, by emphasizing the importance of the nucleoporin Nup85 in inter-complex interactions. Furthermore, we hint at the role of the nucleoporin Nsp1, since instances of this protein type are involved both in copies of the T-complex and copies of the Nup82-complex.

In order to complete these analysis, we also propose tools aiming at comparing two graphs encoding protein contacts, typically one graph coding the contacts in a complex of the toleranced model, and one graph corresponding to an atomic resolution model of this complex (Chapter 6). We formalize this problem as a Maximal Common Sub-graphs problem, and rephrase it in terms of so-called perfect and alternate matchings. Using this machinery for the NPC, we show that despite the wrong location of Sec13 in the toleranced model, it remains possible to recover the totality of contacts foreseen by the atomic resolution model of the Y-complex, except (Nup120, Nup145C) and (Sec13, Nup145C). Our analysis also reveals that ten out of 16 copies of the T-complex found in the toleranced model include the contact (Nic96, Nup49), but not the contact (Nic96, Nup57). This asymmetry has not been noted in biochemistry so far.

From a software standpoint, these analysis were conducted using the VORATOM library, developed during this thesis (see Chapter 7). This C++ library consists of a set of independent tools, aiming at creating and assessing toleranced models from (probability) density maps of protein assemblies. In particular, the computation of compoundly weighted Voronoi diagrams, the analysis of Hasse diagrams and the computation of common sub-graphs can be used in a more general context. Furthermore, the generic design of the C++ classes offers maximal flexibility, since one can change the algorithms or the input/output formats independently from the other components.

Despite these contributions, a number of open questions deserve further work.

First, we have used as input the probability density maps of the individual protein species of the NPC. It would be beneficial to employ density maps from cryo-electron microscopy. The problem of partitioning a map involving a prescribed number of protein instances is a key problem, which could be tackled using information on the shapes and possibly the (relative) position of the proteins. This problem is in general difficult and ill-posed one. A novel approach could consist of using information of the local maxima (and more generally all the critical points) of the map, resorting to Morse theory. But designing noise resilient algorithms in this context is a challenge. Another approach could be to use geometric covering algorithms, to cover selected regions of the map with a prescribed budget of pseudo-atoms.
Second, our construction of toleranced models is based on canonical configurations of molecules. This is a rather elementary strategy which needs to be improved. Given an envelope contoured within a map, approximating this envelope using (toleranced) balls is a possible route. But it is also reminiscent from geometric covering, and thus hard.
Third, the choice of a linear growth process is questionable, since the gradient vector field in a density maps does not generally comply with such a model. Using different growth processes, possibly coupled to anisotropic Voronoi diagrams seems interesting. But the compoundly weighted Voronoi diagram is already quite difficult to handle, and even more elaborate growth processes might be intractable—at least from the symbolic point of view. An alternative could be to use of a linear growth process with respect to the square radii, tantamount to the Mobius diagram, where bisectors are hyper-spheres. But the semantics of such a growth process in the context of a density map remains unclear.

Finally, designing an incremental and output sensitive algorithm for the compoundly weighted Voronoi diagram is a problem which needs to be addressed. This would allow calculations on models of the order of tens of thousands of balls, which are currently beyond reach.

Bibliography

[ADV$^+$07a] F. Alber, S. Dokudovskaya, L. M. Veenhoff, W. Zhang, J. Kipper, D. Devos, A. Suprapto, O. Karni-Schmidt, R. Williams, B.T. Chait, M.P. Rout, and A. Sali. Determining the Architectures of Macromolecular Assemblies. *Nature*, 450(7170):683–694, Nov 2007.

[ADV$^+$07b] F. Alber, S. Dokudovskaya, L. M. Veenhoff, W. Zhang, J. Kipper, D. Devos, A. Suprapto, O. Karni-Schmidt, R. Williams, B.T. Chait, A. Sali, and M.P. Rout. The Molecular Architecture of the Nuclear Pore Complex. *Nature*, 450(7170):695–701, Nov 2007.

[AFK$^+$08] F. Alber, F. Förster, D. Korkin, M. Topf, and A. Sali. Integrating Diverse Data for Structure Determination of Macromolecular Assemblies. *Ann. Rev. Biochem.*, 77:11.1–11.35, 2008.

[BBH01] S.M. Bailer, C. Balduf, and E. Hurt. The Nsp1p Carboxy-Terminal Domain Is Organized into Functionnaly Distinct Coiled-Coil Regions Required for Assembly of Nucleoporin Subcomplexes and Nucleocytoplasmic Transport. *Molecular and Cellular Biology*, 21(23):7944–7955, 2001.

[BG85] W. Braun and N. Gö. Calculation of protein conformations by proton-proton distance constraints: A new efficient algorithm. *Journal of Molecular Biology*, 186:611–626, 1985.

[BLS$^+$08] S.G. Brohawn, N.C. Leksa, E.D. Spear, K.R. Rajashankar, and T.U. Schwartz. Structural evidence for common ancestry of the nuclear pore complex and vesicle coats. *Science*, 322:1369–1373, 2008.

[BS09] S.G. Brohawn and T.U. Schwartz. Molecular Architecture of the Nup84–Nup145C–Sec13 Edge Element in the Nuclear Pore Complex Lattice. *Nat. Struct. Mol. Biol.*, 16(11):1173–1178, 2009.

[BSHP$^+$98] N. Belgareh, C. Snay-Hodge, F. Pasteau, S. Dagher, C.N. Cole, and V. Doye. Functional Characterization of a Nup159p-containing Nuclear Pore Subcomplex. *Molecular Biology of Cell*, 9:3475–3492, 1998.

[BWY06] J.-D. Boissonnat, C. Wormser, and M. Yvinec. Curved voronoi diagrams. In J.-D. Boissonnat and M. Teillaud, editors, *Effective Computational Geometry for curves and surfaces*. Springer-Verlag, Mathematics and Visualization, 2006.

[CCS11] F. Cazals and D. Cohen-Steiner. Reconstructing 3d compact sets. *Computational Geometry Theory and Applications*, 45(1-2):1–13, 2011.

[CD10] F. Cazals and T. Dreyfus. Multi-scale Geometric Modeling of Ambiguous Shapes with Toleranced Balls and Compoundly Weighted α-shapes. In B. Levy and O. Sorkine, editors, *Symposium on Geometry Processing*, Lyon, 2010.

[CK05] F. Cazals and C. Karande. An algorithm for reporting maximal *c*-cliques. *Theoretical Computer Science*, 349(3):484–490, 2005.

[CKL11] F. Cazals, H. Kanhere, and S. Loriot. Computing the volume of union of balls: a certified algorithm. *ACM Transactions on Mathematical Software*, 38(1), 2011.

[CSEH05] D. Cohen-Steiner, H. Edelsbrunner, and J. Harer. Stability of persistence diagrams. In *ACM Symp. Comp. Geometry*, 2005.

[CT09] M. Caroli and M. Teillaud. Computing 3D periodic triangulations. *Algorithms-ESA 2009*, pages 59–70, 2009.

[DMS$^+$08] E.W. Debler, Y. Ma, H.-S. Seo, K.-C. Hsia, T.R. Noriega, G. Blobel, and A. Hoelz. A fence-like coat for the nuclear pore membrane. *Molecular Cell*, 32:815–826, 2008.

[DPU$^+$03] D.P. Denning, S.S. Patel, V. Uversky, A.L. Fink, and M. Rexach. Disorder in the Nuclear Pore Complex: the FG Repeat Regions of Nucleoporins are Natively Unfolded. *PNAS*, 100(5):2450–2455, 2003.

[Ede92] H. Edelsbrunner. Weighted Alpha Shapes. Technical Report UIUCDCS-R-92-1760, Dept. Comput. Sci., Univ. Illinois, Urbana, IL, 1992.

[ELZ02] H. Edelsbrunner, D. Letscher, and A. Zomorodian. Topological persistence and simplification. *Discrete Comput. Geom.*, 28:511–533, 2002.

[Eri09] H.P. Erickson. Size and shape of protein molecules at the nanometer level determined by sedimentation, gel filtration, and electron microscopy. *Biol Proced Online*, 11:32–51, 2009.

[FO10] E. Feliu and B. Oliva. How different from random are docking predictions when ranked by scoring functions? *Proteins*, 78(16), 2010.

[Fra06] J. Frank. *Three-dimensional electron microscopy of macromolecular assemblies: visualization of biological molecules in their native state*. Oxford University Press, USA, 2006.

[GDH93] P. Grandi, V. Doye, and E.C. Hurt. Purification of NSP1 reveals complex formation with'GLFG'nucleoporins and a novel nuclear pore protein NIC96. *The EMBO Journal*, 12(8):3061, 1993.

[GEW$^+$95] P. Grandi, S. Emig, C. Weise, F. Hucho, T. Pohl, and E.C. Hurt. A Novel Nuclear Pore Protein Nup82p Which Specifically Binds to a Fraction of Nsp1p. *The Journal of Cell Biology*, 130(6):1263–1273, 1995.

[Hal04] R.A. Hall. Studying protein-protein interactions via blot overlay or far western blot. *METHODS IN MOLECULAR BIOLOGY-CLIFTON THEN TOTOWA-*, 261:167–174, 2004.

[HC95] S.E. Harding and H. Cölfen. Inversion Formulae for Ellipsoid of Revolution Macromolecular Shape Functions. *Analytical Biochemistry*, 228:131–142, 1995.

[HdcB98] M.E. Hurwitz, C. Strambio de castillia, and G. Blobel. Two yeast nuclear pore complex proteins involved in mRNA export form a cytoplasmically oriented complex. *PNAS*, 95(19):11241–11245, 1998.

[HGC94] Y. Harpaz, M. Gerstein, and C. Chothia. Volume Changes on Protein Folding. *Structure*, 2:641–649, 1994.

[HSBH07] K.-C. Hsia, P. Stavropoulos, G. BLobel, and A. Hoelz. Architecture of a coat for the nuclear pore membrane. *Cell*, 131(7):1313–1326, 2007.

[JS07] S. Jeudy and T.U. Schwartz. Crystal structure of nucleoporin Nic96 reveals a novel, intricate helical domain architecture. *J. Biol. Chem.*, 282(1):34904–34912, 2007.

[KB09] M. Kampmann and G. Blobel. Three-Dimensional Structure and Flexibility of a Membrane-Coating Module of the Nuclear Pore Complex. *Nat. Struct. Mol. Biol.*, 16(7):782–788, 2009.

[Koc01] I Koch. Fundamental study: Enumerating all connected maximal common subgraphs in two graphs. *Theoretical Comp. Sc.*, 250(1-2):1–30, 2001.

[LBS09] N.C. Leksa, S.G. Brohawn, and T.U. Schwartz. The structure of the scaffold nucleoporin nup120 reveals a new and unexpected domain architecture. *Structure*, 17:1082–1091, 2009.

[LTSW09] K. Lasker, M. Topf, A. Sali, and H.J. Wolfson. Inferential optimization for simultaneous fitting of multiple components into a cryoem map of their assembly. *Journal of molecular biology*, 388(1):180–194, 2009.

[LW10] M.F. Lensink and S.J. Wodak. Docking and scoring protein interactions: Capri 2009. *Proteins: Structure, Function, and Bioinformatics*, 78:3073–3084, 2010.

[M. 11] M. Kampmann and C.E. Atkinson and A.L. Mattheyses and S.M. Simon. Mapping the Orientation of Nuclear Pore Proteins in Living Cells with Polarized Fluorescence Microscopy. *Nat. Struct. Mol. Biol.*, 18(6):643–652, 2011.

[NHD$^+$09] V. Nagy, K.-C. Hsia, E.W. Debler, M. Kampmann, A.M. Davenport, G. Blobel, and A. Hoelz. Structure of a trimeric nucleoporin complex reveals alternate oligomerization states. *PNAS*, 106(42):17693, 2009.

[OBSC00] A. Okabe, B. Boots, K. Sugihara, and S. N. Chiu. *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams (2nd Ed.)*. Wiley, 2000.

[PCR$^+$01] O. Puig, F. Caspary, G. Rigaut, B. Rutz, E. Bouveret, E. Bragado-Nilsson, M. Wilm, and B. Séraphin. The tandem affinity purification method: A general procedure of protein complex purification. *Methods*, 24:218–229, 2001.

[SH01] H. Schwartz and H. Hohenberg. *Immuno-electron Microscopy*. Wiley Online Library, 2001.

[SHL$^+$97] N.L. Schlaich, M. Haner, A. Lustig, U. Aebi, and E.C. Hurt. In vitro reconstitution of a heterotrimeric nucleoporin complex consisting of recombinant Nsp1p, Nup49p, and Nup57p. *Molecular biology of the cell*, 8(1):33–46, 1997.

[SMD$^+$04] H.-S. Seo, Y. Ma, E.W. Debler, D. Wacker, S. Kutik, G. Blobel, and A. Hoelz. Structural and functional analysis of nup133 domains reveals modular building blocks of the nuclear pore complex. *J. Cell Biol.*, 167:591–597, 2004.

[SMD$^+$09] H.-S. Seo, Y. Ma, E.W. Debler, D. Wacker, S. Kutik, G. Blobel, and A. Hoelz. Structural and Functional Analysis of Nup120 Suggests Ring Formation of the Nup84 Complex. *PNAS*, 106(34):14281–14286, 2009.

[SSF$^+$08] N. Schrader, P. Stelter, D. Flemming, R. Kunze, E. Hurt, and I.R. Vetter. Structural Basis of the Nic96 Subcomplex Organization in the Nuclear Pore Channel. *Molecular Cell*, 29(1):46–55, 2008.

[WR10] S.R. Wente and M.P. Rout. The Nuclear Pore Complex and Nuclear Transport. *Colde Spring Harbor Perspectives in Biology*, 2(10):a000562, 2010.

[WS09] J.R.R. Whittle and T.U. Schwartz. Architectural nucleoporins nup157/170 and nup133 are structurally related and descend from a second ancestral element. *J. Biol. Chem.*, 284:28442–28452, 2009.

Communications

Publications

Multi-scale Geometric Modeling of Ambiguous Shapes with Toleranced Balls and Compoundly Weighted alpha-shapes;
Frédéric Cazals, Tom Dreyfus;
Computer Graphics Forum (SGP) 2010 29(5): 1713–1722

Assessing the Reconstruction of Macro-molecular assemblies with Toleranced Models;
Tom Dreyfus, Valérie Doye and Frédéric Cazals;
Submitted

Probing a Continuum of Macro-molecular Assembly Models with Graph Templates of Sub-complexes;
Frédéric Cazals and Tom Dreyfus;
Submitted

An Incremental Algorithm to Compute Compoundly Weighted Voronoi Diagrams;
Frédéric Cazals and Tom Dreyfus;
In preparation

Invited Presentations at International Conferences

Assessing the Stability of Protein Complexes within Large Assemblies;
EMBO Symposium on Molecular Perspectives on Protein-Protein Interactions,
Sant Feliu de Guixols, Spain, November 2010

Assessing the Stability of Protein Complexes within Large Assemblies;
17th International Biophysics Congress,
Beijing, China, November 2011.

Invited Presentations at National Conferences

Assessing the Stability of Protein Complexes within Large Assemblies;
XXIIeme Congrès de la Société Française de Biophysique,
La-Colle-sur-Loup, France, September 2010.

Multi-scale Analysis of Uncertain Data: the λ-complex Fitlration;
Journées de Géométrie Algorithmique,
Marseille, France, March 2010.

Posters at National Conferences

Improving the Reconstruction of Large Macro-Molecular Assemblies: the Example of the Nuclear Pore Complex (NPC);
EMBO workshop, Barcelona, Spain, October 2008.
Assessing the Reconstruction of Macro-molecular Assemblies: the Example of the Nuclear Pore Complex;
GGMM, La Rochelle, France, June 2011.

RÉSUMÉ

La génomique structurale a donnée accès à un nombre remarquable d'informations sur le protéome. De nature essentiellement combinatoire—il apparaît que certaines protéines interagissent en complexe, elles gagnent à être complémentées par des modèles tridimensionnels pour étendre la connaissance jusqu'au niveau structural. Récemment, de tels modèles ont été reconstruits pour le pore nucléaire, en intégrant diverses données biophysiques et biochimiques. Cependant, la nature qualitative de ces modèles empêche une complète synergie entre ceux-ci et les données expérimentales. Cette thèse propose trois développements répondant à ces limitations.

Premièrement, nous introduisons les modèles tolérancés pour représenter des formes aux contours incertains par un continuum de modèles. Nous montrons qu'un modèle tolérancé est équivalent à un diagramme de Voronoï additif multiplicatif, et nous développons le λ-complexe, l'équivalent de l'α-complexe, pour un tel diagramme.
Deuxièmement, nous utilisons les modèles tolérancés pour représenter des assemblages protéiques. Nous expliquons comment un modèle tolérancé peut être utilisé pour évaluer la stabilité des contacts entre les protéines et pour valider la cohérence d'un tel modèle vis à vis de données expérimentales.
Troisièmement, nous proposons des outils pour comparer des graphes de contact entre protéines, issus d' une part d'un modèle tolérancé, et d'autre part d'un modèle connu à résolution atomique.
L'ensemble de ces concepts et outils est utilisé pour sonder les reconstructions du pore nucléaire mentionnées ci-dessus.

Mots-clés. Diagramme de Voronoï courbe; Complexe de Delaunay; α-complexe; Modélisation avec incertitudes; Complexes protéiques; Assemblages macromoléculaires; Pore nucléaire; Évaluation de modèles

ABSTRACT

Structural genomics projects have revealed remarkable features of proteomes. But these are essentially of combinatorial nature—selected proteins interact within a complex, so that extending them to the structural level requires building 3D models of these complexes. Such models have recently been reconstructed for the Nuclear Pore Complex (NPC), based on the integration of diverse biophysical and biochemical data. Yet, a full synergy between them and the experimental data is not at play because the reconstructions are qualitative. This thesis makes three contributions addressing these limitations.

First, we introduce toleranced models (TOM) to inherently represent uncertain shapes as a continuum of models. We show that a TOM is equivalent to a compoundly weighted Voronoi diagram, and develop the λ-complex, the equivalent of the α-complex for such a diagram.
Second, we use toleranced models to represent protein assemblies. We explain how a toleranced model can be used to assess stable contacts between proteins and to check the coherence between such a model and experimental data.
Third, we propose tools to compare graphs encoding contacts within proteins, such graphs coming from a toleranced model on the one hand, and from atomic-resolution models on the other hand.
All these concepts and tools are used to probe the aforementioned reconstructions of the NPC.

Key-words. Curved Voronoi diagram; Delaunay complex; α-complex; Uncertain models; Protein complexes; Macromolecular assemblies; Nuclear pore complex; Model assessment