

N° d'ordre :

Réification des interactions
pour l'expérience *in virtuo*
de systèmes biologiques multi-modèles.

Thèse

présentée devant

l'Université de Bretagne Occidentale

pour obtenir le grade de :

DOCTEUR DE L'UNIVERSITÉ DE BRETAGNE OCCIDENTALE
Mention INFORMATIQUE

par

Gireg DESMEULLES

Équipe d'accueil 3883 :

Laboratoire d'Informatique des Systèmes Complexes (LISyC/UBO-ENIB)
Centre Européen de Réalité Virtuelle (CERV)

Soutenue le 11 décembre 2006 devant la commission d'examen :

Vincent	CHEVRIER	Maître de Conférences, HdR, LORIA, Nancy	<i>Rapporteur</i>
Jean-Pierre	JESSEL	Professeur, IRIT, Toulouse	<i>Rapporteur</i>
Marie	BEURTON-AIMAR	Maître de Conférences, LaBRI, Bordeaux	<i>Examineur</i>
Laurent	MISERY	Professeur, CHU-UBO, Brest	<i>Examineur</i>
Vincent	RODIN	Professeur, UBO, Brest	<i>Encadrant</i>
Jacques	TISSEAU	Professeur, ENIB, Brest	<i>Directeur</i>

**Réification des interactions
pour l'expérience in virtuo
de systèmes biologiques multi-modèles**

thèse de doctorat

GIREG DESMEULLES

**Laboratoire d'Informatique des Systèmes Complexes
EA3883 (UBO, ENIB)**

Gireg DESMEULLES
tel : +33 (0)2 98 05 89 53
fax : +33 (0)2 98 05 89 79
courriel : desmeulles@enib.fr

Centre Européen de Réalité Virtuelle

25 rue Claude Chappe
BP 38
F-29280 Plouzané
tel : +33 (0)2 98 05 89 89
fax : +33 (0)2 98 05 89 79
courriel : contact@cerv.fr
internet : <http://www.cerv.fr>



Remerciements

Je tiens à remercier les nombreuses personnes qui par leurs conseils et par leur soutien m'ont permises de mener à bien cette thèse. J'ai sollicité tant de personnes durant ce travail qu'il m'est difficile d'être exhaustif, je présente par avance mes excuses aux « oubliés ».

Je remercie donc :

JEAN-PIERRE JESSEL et VINCENT CHEVRIER pour avoir rapporté cette thèse et MARIE BEURTON-AIMAR pour avoir bien voulu être examinatrice.

JACQUES TISSEAU, le directeur de cette thèse qui m'a accueilli au sein de son laboratoire, m'offrant ainsi la chance d'être témoin de la création du Centre Européen de Réalité Virtuelle, et le plaisir de m'appropriier certaines des idées originales qui y circulent.

VINCENT RODIN pour son encadrement et pour avoir initié avec LAURENT MISERY le projet de collaboration sur lequel repose cette thèse.

L'ensemble des membres de l'équipe de neurobiologie cutanée du CHU de Brest, en particulier LAURENT MISERY, ULYSSE PEREIRA, YANNICK CHATEAU, NICHOLAS BOULAIS.

NICOLAS FROMENT pour avoir été un collaborateur agréable et pour avoir largement contribué à la mise en œuvre de la partie applicative de cette thèse.

La team Bio, PATOCHE, PIT, GABY, CHAMPION, GOBI ET MICHAEL qui ont chacun participé d'une manière ou d'une autre à ce travail.

Les void* sans qui ça ne compile pas.

Tous les gens du CERV pour leur disponibilité, leurs conseils et l'ambiance de travail très agréable qu'ils y créent.

Enfin je remercie mes nombreux relecteurs orthographiques (merci ln, papa, maman...).

Financement

Ce travail a bénéficié du soutien financier de la Région Bretagne.



Résumé

Réification des interactions pour l'expérience *in virtuo* de systèmes biologiques multi-modèles.

Résumé

Cette thèse s'inscrit dans le cadre de la modélisation de phénomènes biologiques pour leur expérimentation à travers un système de réalité virtuelle. Son objet est la définition d'un cadre générique de modélisation et d'implémentation adapté à l'étude des systèmes physiologiques.

En premier lieu, le modèle générique proposé s'appuie sur le principe de la réification des interactions en objets actifs autonomes. Ensuite, il permet l'organisation des modèles biologiques en un agencement de systèmes autonomes. Il rassemble alors deux conceptions de l'autonomie : l'une est destinée à concevoir les systèmes de réalité virtuelle et l'autre a pour objet la modélisation en biologie.

Le modèle générique est dérivé en un certain nombre d'outils de modélisation pour la biologie. La bibliothèque composée du modèle générique et des outils de modélisation permet alors de réaliser différentes applications. La principale application a pour objet la mise en œuvre d'un modèle du phénomène d'urticaire allergique. Enfin, un modèle de système autopoïétique minimal est proposé pour illustrer les possibilités de la méthode.

Mots clés

Simulation, multi-agents, réalité virtuelle, *in virtuo*, biologie intégrative, interaction.

Interactions reification to experience biological multi-models systems *in virtuo*.

Abstract

To model biological systems and experiment them through a virtual reality application is the purpose of this thesis. The aim is to provide the definition of a generic modelling framework and its implementation for the study of physiological systems.

In the first place, the generic model is based on the reification of interactions into autonomous active objects. Thereby, the biological models can be organized in a layout of autonomous systems. Therefore, the generic model infers two conceptions of autonomy : the first one is used to design virtual reality systems and the second one is oriented towards biological modelling.

The generic model is specialized into several modelling tools for biology. Thereafter, the library composed by the generic models and the tools allows the building of applications. The purpose of the main application is to implement the model of an allergic urticaria phenomenon. At last, the model of a minimal autopoietic system exemplifies the method's potentials.

Keywords

Simulation, multi agent, Virtual reality, *in virtuo*, systems biology, interaction.

Table des matières

Remerciements	iii
Financement	v
Résumé	vii
Table des matières	xiii
Table des figures	xvii
Introduction	1
Positionnement de la thèse	1
Organisation du mémoire	3
I État de l’art	5
1 La complexité des biologistes	7
1.1 La biologie, une science sans théorie	7
1.2 La biologie moléculaire, l’étude de l’ordre qui engendre l’ordre	8
1.3 Le génie génétique et l’explosion des données	9
1.4 Bio-informatique	9
1.4.0.1 Gestion et Organisation des données	9
1.4.0.2 Comparer, analyser, explorer les informations	10
1.5 Changement de méthode : biologie systémique, théorique ou intégrative	11
1.5.0.3 Élargir le champ d’application : les scopes	13
2 Complexité biologique et informatique	15
2.1 Des mathématiques pour l’étude de la complexité	15
2.1.1 Le calcul infinitésimal	15
2.1.2 Les systèmes complexes	17
2.1.2.1 Les systèmes dynamiques	17
2.1.2.2 Systèmes non-linéaires	20
2.1.2.3 Les bifurcations	20

2.2	Les formalismes « bio-logiques »	21
2.2.1	La complexité des réseaux	22
2.2.2	Les algèbres de processus	24
2.2.3	Les réseaux de Petri	24
2.2.4	Les logiques temporelles	26
2.2.5	Conclusions	27
2.3	L'avènement de la simulation numérique	28
2.3.1	L'expérimentation <i>in vivo</i>	28
2.3.2	L'expérimentation <i>in vitro</i>	29
2.3.3	Le raisonnement <i>in abstracto</i>	29
2.3.4	Les calculs <i>in silico</i>	29
2.3.5	La granularité des modèles	30
2.3.5.1	Microscopique	30
2.3.5.2	Macroscopique	31
2.3.5.3	Mésoscopique	31
2.3.6	Vers une simulation multi-échelles	32
2.4	Types d'architectures pour les simulations <i>in silico</i>	32
2.4.1	La simulation « multi-modèles »	33
2.4.2	SBML	34
2.4.2.1	La « multi-simulations »	34
2.4.2.2	La norme	34
2.4.2.3	Éditeurs graphiques	35
2.4.3	L'exemple de Virtual Cell	35
2.4.4	Architectures orientées objet	39
2.4.4.1	UML pour formaliser le vivant	40
2.4.5	SMA et automates cellulaires	43
2.4.5.1	Paradigmes individus-centrés	43
2.4.5.2	Applications	44
2.5	Conclusion	46
3	Complexité biologique, informatique et cognition	49
3.1	Cognition, expérience et science	49
3.1.1	La cognition science de la complexité	50
3.1.2	Le rôle de l'expérience	52
3.2	Réalité virtuelle et biologie : Quels concepts ?	53
3.2.1	Étymologie	53
3.2.2	Définitions	54
3.2.3	L'expérience <i>in virtuo</i>	55
3.2.4	...pour la biologie.	57
3.3	Le virtuoscope : Quels outils ?	58
3.3.1	Les travaux du Centre Européen de Réalité	58
3.3.2	L'autonomie	59
3.3.2.1	L'autonomie dans les univers virtuels	59
3.3.2.2	La médiation de l'esprit	60
3.4	Le projet In Virtuo : Quels modèles ?	60
3.4.1	Réalité virtuelle et biologie	60
3.4.2	Modèles microscopiques : agent Cellule et agent Molécule	61
3.4.2.1	Construction d'un agent Cellule	62
3.4.2.2	Comportement d'un agent Cellule	63

3.4.2.3	Interactions entre agents Cellule , agents Molécule	63
3.4.2.4	Intérêt et utilisation des agents Cellule	64
3.4.3	Agents Réaction et Diffusion	64
3.4.3.1	Historique	64
3.4.3.2	Principe de fonctionnement	64
3.4.3.3	Formalisation et validation	65
3.4.3.4	La diffusion	66
3.4.4	Agents Organisation	67
3.5	Le besoin d'une architecture générique	69
 II Modèles		 71
4	Le modèle générique	73
4.1	Réification des interactions	73
4.1.1	Motivations	73
4.1.2	Abstraction des composants	74
4.1.3	Point de départ du modèle informatique	75
4.1.4	Agents ou objets actifs ?	76
4.2	L'ordonnancement asynchrone multi-échelles	77
4.2.1	L'ordonnancement	77
4.2.1.1	La notion de cycle	77
4.2.1.2	Temps réel et temps virtuel	77
4.2.1.3	L'hypothèse de synchronicité	78
4.2.2	Ordonnancement asynchrone chaotique sans remise	80
4.2.2.1	Introduction de la composante stochastique	80
4.2.2.2	Introduction du multi-échelles	80
4.2.3	Discussion	83
4.2.3.1	Asynchrone synchronisé ?	83
4.2.3.2	Ordonnancement avec remise	84
4.2.4	Solution technique adoptée	85
4.2.5	Limite de la composante stochastique	86
4.3	Organisations multi-niveaux	87
4.3.1	L'organisation	87
4.3.2	L'organisation statique	88
4.3.3	Les phénomènes	88
4.3.4	Les organisations	90
5	Dérivation du modèle pour la biologie	95
5.1	Les phénomènes chimiques	95
5.1.1	L'interaction Réaction	95
5.1.2	L'interaction Diffusion	97
5.1.3	L'organisation modèle SBML	98
5.1.4	Organisation chimique	100
5.2	Organisation de la cellule	101
5.2.1	La cellule chimique	101
5.2.2	Couplage de la cellule avec le milieu	101
5.3	Organisation mécanique	103
5.3.1	Introduction de modèles mécaniques	103
5.3.2	Optimisations	104

5.3.3	Interaction ponctuelle, cas limite d'utilisation	105
5.4	L'humain dans la boucle	106
5.4.1	L'utilisateur : médiation des sens et de l'action	106
5.4.2	Le modélisateur : médiation de l'esprit	107
5.4.3	À propos des interfaces	108
5.5	La libBio	109
III Applications		111
6	Derme <i>in virtuo</i>	113
6.1	Contexte biologique.	113
6.2	Intérêts mutuels du projet	114
6.3	L'organisation Derme	115
6.3.1	Le milieu	115
6.3.2	L'épiderme	116
6.3.3	La membrane basale	117
6.4	Les mastocytes	118
6.4.1	Description	118
6.4.2	Relargage d'histamine et estimation de paramètres	119
6.4.3	Visualisation	120
6.5	Capillaire sanguin	121
6.6	Fibre nerveuse	122
6.7	Les interfaces	124
6.7.1	Visualisation et interaction 3D	124
6.7.2	Sonde	126
6.7.3	Seringue	126
6.8	Discussions	128
7	Figures de style	131
7.1	Un autre modèle de diffusion	131
7.1.1	Le problème de la structuration <i>a priori</i>	131
7.1.2	Maillage par plans médians	132
7.1.3	Intégration dans le cadre générique	133
7.1.4	Solution technique : diagramme de Voronoï	134
7.1.5	Formalisation du calcul de l'interaction Diffusion	136
7.1.6	Perspectives	137
7.2	Un système autopoïétique minimal	137
7.2.1	Systèmes vivants et systèmes du vivant	137
7.2.2	L'autopoïèse	138
7.2.3	Système proposé	138
7.2.3.1	Structure dissipative	138
7.2.3.2	La brique élémentaire MyCell	139
7.2.3.3	L'organisation AutopoieticOrganisation	141
7.2.4	Expérimentation	143
7.2.5	Discussion	145
Conclusions		147
	Bilan	147
	Perspectives	149

Annexes	155
A Mast.xml	155
B Diagrammes de classes UML	165
C Résumé long	169
Références bibliographiques	184

Table des figures

1	La complexité des biologistes	7
2	Complexité biologique et informatique	15
2.1	Attracteurs périodique et ponctuel	20
2.2	Attracteur de Lorenz	21
2.3	Diagramme de bifurcation	22
2.4	Graphes complexes	23
2.5	π -calcul	25
2.6	Réseau de Petri	26
2.7	Réaction enzymatique SBML	36
2.8	Bio Sketch Pad	37
2.9	CellDesigner	37
2.10	Virtual Cell	38
2.11	Virtual Cell 2	39
2.12	CellAK	41
2.13	CellAK 2	42
2.14	Cellulat	46
3	Complexité biologique, informatique et cognition	49
3.1	Les différents types de modèles	56
3.2	Les différents modes d'experimentation	57
3.3	Autonomie interaction immersion	59
3.4	Création d'un agent <code>Cellule</code> vide	62
3.5	Création d'un agent <code>Cellule</code> complet	62
3.6	Comportements de base prédéfinis	63
3.7	Principe des agents <code>Réaction</code>	65
3.8	Principe des agents <code>Diffusion</code>	67
3.9	Tumeur	68
3.10	Voies de signalisation intra-cellulaires	68
3.11	Diagramme de classes introduisant la notion d'organisation	68

4	Le modèle générique	73
4.1	Graphes duaux	75
4.2	Réification des interactions UML 1	75
4.3	Réification des interactions UML 2	76
4.4	Ordonnements synchrone et asynchrone	81
4.5	Ordonnement chaotique sans remise	82
4.6	Ordonnement chaotique sans remise et multi-échelles	83
4.7	Ordonneur avec remise	85
4.8	Ordonneur UML	86
4.9	Organisation UML 1	88
4.10	Organisation UML 2	89
4.11	Couplage structurel	91
4.12	Organisation UML 3	92
5	Dérivation du modèle pour la biologie	95
5.1	L'organisation réacteur	96
5.2	Réaction chimique UML :	97
5.3	Diffusion	98
5.4	Diffusion UML	99
5.5	Modèle SBML	99
5.6	Organisation Chimique	100
5.7	Cellule	102
5.8	Collisions	103
5.9	Collisions UML	104
5.10	Collisions dans une tumeur	105
5.11	La libBio	110
6	Derme <i>in virtuo</i>	113
6.1	Capture d'écran de l'environnement	115
6.2	Visualisation des corps des kératinocytes	117
6.3	Kératinocyte UML	117
6.4	Membrane basale UML	117
6.5	Vue de la membrane basale	118
6.6	Visualisation du corps des mastocytes	119
6.7	Les deux voies principales d'activation du mastocyte	120
6.8	Représentation 3D des différents états des mastocytes	121
6.9	Capillaire UML	122
6.10	Vue du capillaire sanguin	123
6.11	Fibre nerveuse UML	123
6.12	Vue de l'activation de la fibre nerveuse	124
6.13	Viewer3D d'ARéVi	125
6.14	Vue de la sonde	126
6.15	Expérimentation <i>in virtuo</i> du modèle de derme	127
7	Figures de style	131
7.1	La taille du maillage	132
7.2	Plans médians 1	133
7.3	Plans médians 2	133
7.4	Organisation chimique Voronoï	134
7.5	Organisation chimique Voronoï 2	135

7.6	Delaunay et Voronoï	135
7.7	Déplacement et modification du maillage	136
7.8	Environnement	139
7.9	Source d'énergie	140
7.10	MyCell UML	141
7.11	MyCell	142
7.12	AutopoieticOrganization UML	142
7.13	Évolution du système autopoïétique	143
7.14	Amas cellulaire	144
Annexes		155
A Mast.xml		155
B Diagrammes de classes UML		165
B.1	Détail des classes de base de la libBio	166
B.2	Classes liées au phénomènes de nature chimique	167
B.3	Classe liées au phénomènes de colisions	168
C Résumé long		169
C.1	Diagramme UML du modèle générique	171
C.2	Capture d'écran de l'application	172

Introduction

« *Le modèle de la première cybernétique est un modèle où l'autonomie n'existe pas. Il contraste parce que les inputs et les outputs sont les points centraux. C'est par définition un système hétéronome, c'est l'élément heteros (l'autre) qui est fondamental et non la boucle autos (soi-même) qui particularise le système* » Francisco Varéla [Benkirane, 2002].

Positionnement de la thèse

Cette thèse s'inscrit dans le cadre de la modélisation de phénomènes biologiques pour leur expérimentation à travers un système de réalité virtuelle. Son objet est la définition d'un cadre générique de modélisation et d'implémentation adapté à l'étude des systèmes physiologiques.

Les systèmes considérés appartiennent en premier lieu à la catégorie des systèmes complexes. Cette complexité provient avant tout de la diversité des composants, de la diversité des structures et de la diversité des interactions mises en jeu. Un tel système est alors *a priori* un milieu ouvert (apparition/disparition dynamique de composants), hétérogène (morphologies et comportements variés) et formé d'entités composites, mobiles et distribuées dans l'espace, en nombre variable dans le temps. Ces composants, parmi lesquels l'homme avec son libre arbitre joue souvent un rôle déterminant, peuvent être structurés en différents niveaux connus initialement ou émergents au cours de leur évolution du fait des multiples interactions entre ces composants. Les interactions elles-mêmes peuvent être de natures différentes et opérer à différentes échelles spatiales et temporelles.

Il n'existe pas aujourd'hui de théorie capable de formaliser cette complexité et de fait, il n'existe pas (ou peu) de méthodes de preuves formelles *a priori* comme il en existe dans les modèles hautement formalisés. En l'absence de preuves formelles, nous devons recourir à l'expérimentation du système en cours d'évolution afin de pouvoir effectuer des validations expérimentales *a posteriori*.

La réalité virtuelle (RV) fournit aujourd'hui un cadre conceptuel, méthodologique et expérimental bien adapté pour imaginer, modéliser et expérimenter cette complexité. L'utilisateur d'un système de réalité virtuelle, immergé en temps réel dans cet espace par la triple médiation des sens (perception), de l'action (expérimentation), et de l'esprit (modélisation) peut être spectateur, acteur et/ou créateur [Fuchs et al., 2003].

On parle alors de virtuoscope pour désigner ce laboratoire virtuel pour l'étude des systèmes complexes, qui s'appuie sur les concepts, les modèles et les outils de la réalité virtuelle. La réalité virtuelle place l'utilisateur au cœur du laboratoire virtuel et se rapproche ainsi des méthodes des sciences expérimentales, tout en lui donnant accès aux méthodes numériques. Ce mode d'investigation se nomme l'expérimentation *in virtuo*.

Le virtuoscope est un outil conceptuel, qui repose sur une vision particulière de la RV qui met en exergue le principe d'**autonomie** [Tisseau, 2001]. L'autonomie est ici, l'autonomie des modèles numériques qui peuplent l'univers virtuel et lui permettent d'être régi par ses propres lois.

L'outil conceptuel virtuoscope autorise donc l'expérimentation d'un système complexe, en dotant l'univers virtuel des lois qui permettent la réalisation de ce système. Par exemple, la manipulation d'un système physique est directement envisageable en incarnant les lois physiques au sein du système de réalité virtuelle. Cependant, en ce qui concerne la biologie, science expérimentale, l'existence de lois est beaucoup moins évidente qu'en sciences physiques. Le virtuoscope doit alors être précisé pour s'adapter aux propriétés particulières des systèmes appartenant au vivant. En cela, les travaux de F.Varéla sur la modélisation du vivant nous ont semblé remarquablement adaptés à notre propos. En effet avec l'aide de H.Maturana, il pose la question : « qu'est ce que l'**autonomie** de l'organisme ? ». À partir de l'idée de l'autonomie, un certain nombre de concepts permettant de rendre intelligible l'organisation des mécanismes biologiques ont été construits. L'ouvrage « Autonomie et connaissance » [Varela, 1989a] rassemble ces idées parmi lesquelles la clôture opérationnelle, le couplage structurel et surtout l'autopoïèse.

Le propos de la thèse peut alors se réduire à l'exercice suivant : rassembler au sein d'un modèle générique les deux conceptions de l'autonomie ; l'une étant destinée à concevoir les systèmes de réalité virtuelle et l'autre ayant pour objet la modélisation en biologie. Le premier principe d'autonomie nous conduit alors à modéliser les phénomènes, puis plus particulièrement, les interactions. Les interactions ainsi concrétisées sont rendues autonomes et se faisant, elle rendent l'univers autonome. L'idée de la réification des interactions se prête alors idéalement au second principe d'autonomie. Il devient dès lors possible d'organiser les systèmes en un assemblage de systèmes autonomes en couplage structurel.

Depuis quelques années se forme une communauté de chercheur utilisant l'outil informatique pour formaliser, modéliser et simuler les systèmes complexes de la biologie. L'article de H.Kitano paru dans la revue *Nature* : « *Computational Systems Biology* » [Kitano, 2002] regroupe ces travaux sous l'appellation *systems biology*. Cette recherche interdisciplinaire met en avant le principe des calculs *in silico*. Le concept de l'expérimentation *in virtuo* et le changement de méthode provoqué par la posture originale de modélisation par réification des interactions, amènent à suivre une voie inexplorée par la méthode *in silico*. Nous le verrons, le cadre proposé permet d'appréhender des systèmes composés de modèles de natures variées, agissant à des échelles de temps et à des niveaux de description différents. Elle permet intrinsèquement de considérer une approche multi-modèles ; ce qui représente un des défis actuellement présentés aux outils de simulation *in silico*.

Le modèle générique proposé est alors dérivé en une bibliothèque offrant des facilités pour mettre en œuvre des phénomènes couramment utilisés lors de la modélisation de systèmes physiologiques humains. Ensuite, cette boîte à outils doit être expérimentée au sein d'applications biologiques. Dans ce travail, la collaboration avec l'équipe de neurobiologie cutanée de

l'unité de Physiologie Comparée Intégrative (EA3879) représente le cadre applicatif essentiel. Seule la confrontation aux contraintes inhérentes à la réalisation d'un tel projet peut fournir une forme de validation à la méthode préconisée.

Organisation du mémoire

Ce mémoire s'organise de manière conventionnelle autour de trois grandes parties : état de l'art (partie I), modèles (partie II) et applications (partie III).

1. La partie I : **état de l'art**, définit le contexte de ce travail appliquant successivement les points de vues de trois disciplines sur l'étude des systèmes biologiques. Ainsi nous traitons de la prise en compte de la complexité par la biologie (Ch.1), du traitement de cette complexité en informatique (Ch.2) et des possibilités d'expérience offerte par une prise en compte de la cognition dans la manipulation des modèles (Ch.3).

La complexité des biologistes Il s'agit de donner quelques repères pour situer ce travail d'informaticien dans le contexte de la biologie d'aujourd'hui. Cette science expérimentale possède à l'origine assez peu d'outils théoriques, les rares présents étant empruntés aux autres domaines scientifiques. La génétique a conduit à l'émergence de la biologie moléculaire et à l'explosion de la quantité de données. C'est à ce moment que l'informatique est entrée en scène avec la bio-informatique. Les nouvelles questions posées par ces évolutions ont donné naissance à un nouveau paradigme en biologie répondant aux appellations biologie théorique, systémique ou intégrative. Ce paradigme repose sur la prise en compte de la complexité biologique et fournit de nouveaux défis à la recherche en biologie.

Complexité biologique et informatique Comment l'outil informatique peut-il aider les biologistes à traiter cette complexité inhérente aux systèmes biologiques ? Des méthodes formelles à la simulation, un large panel de voies est aujourd'hui exploré pour manipuler les modèles de ces systèmes. La voie de la simulation est celle que nous privilégions. Elle propose un nouveau moyen d'investigation : les calculs *in silico*. Plusieurs méthodes mettent en œuvre ces calculs. Nous préférons alors les méthodes orientées objets et agents car elles permettent la simulation multi-échelles et multi-modèles plus facilement que les outils classiques reposant sur la résolution d'équations différentielles.

Complexité biologique, informatique et cognition Il est possible d'aller plus loin dans l'acte d'expérience revendiqué par les concepteurs d'expérimentation *in silico*. En effet, le principe de l'énaction nous fournit quelques concepts théoriques pour comprendre et définir la cognition. Les concepts de la réalité virtuelle et le lien fort que ce domaine entretient avec l'acte d'expérience nous permettent alors d'envisager un nouveau type d'expérimentation des modèles numériques, l'expérience *in virtuo*. Cette dernière passe nécessairement par la double médiation des sens et de l'action. Le principe d'autonomie se révèle alors incontournable pour construire les univers virtuels. Se faisant, la réalité virtuelle offre également un troisième type de médiation : la médiation de l'esprit. Différents travaux réalisés au Centre Européen de Réalité Virtuelle tendent ainsi à mettre en pratique les concepts et les outils de la réalité virtuelle pour modéliser et expérimenter les systèmes complexes du vivants. Aujourd'hui, le besoin d'une architecture logicielle permettant de rassembler et de réutiliser ces travaux se fait ressentir.

2. La partie II : **modèles**, présente d'abord le modèle générique proposé (Ch.4) puis sa dérivation en une « boîte à outils » pour la modélisation de systèmes biologiques (Ch.5).

Le modèle générique Le modèle générique constitue le cœur de ce travail, il porte sur trois points essentiels. Premièrement, la réification des interactions est l'idée originale autour de laquelle est construit le modèle. Deuxièmement, l'ordonnement des interactions rendues autonomes doit être défini. Troisièmement, la notion d'organisation permet de définir des systèmes possédant la propriété d'autonomie dans l'univers virtuel.

Dérivation du modèle pour la biologie En dérivant le modèle générique, il est possible de définir un certain nombre de phénomènes réutilisables pour la modélisation biologique. En plus de fournir une boîte à outils pour le modélisateur, ce travail permet d'illustrer l'utilisation du modèle générique. Ainsi, une implémentation est proposée pour les phénomènes de nature chimique. Cette implémentation s'appuie sur la norme SBML. Ensuite, une organisation autonome de la cellule est définie. Une solution pour la gestion des collisions est également décrite. Nous introduisons enfin le modèle particulier de l'utilisateur.

3. La partie III : **applications**. La « boîte à outils » produite est alors utilisée dans deux types d'application. Premièrement, l'application à un cas concret d'étude, permet de se confronter aux réalités de la recherche en biologie (Ch.6). Puis, deux applications originales permettent d'ouvrir quelques perspectives supplémentaires de modélisation (Ch.7).

Application « Derme *in virtuo* » La collaboration avec une équipe de biologistes est le substrat sur lequel s'est construite cette thèse. La modélisation du phénomène de la réaction allergique intervenant lors de l'urticaire allergique a permis de mettre en œuvre des modèles de peau, de mastocyte, de vaisseau capillaire et de fibre nerveuse. Des exemples d'interfaces ont également pu être définis.

Figures de style La première application vise à proposer une méthode adaptative de maillage de l'espace pour les modèles de réaction-diffusion. Nous montrons ainsi comment les modèles structurent eux-mêmes leur environnement. La seconde application est un exercice qui consiste à définir un système autopoïétique minimal. En effet, de l'autonomie à l'autopoïèse, il n'y a qu'un pas. La réalisation d'un tel système ouvre alors de nouvelles perspectives tout en posant de nouvelles questions.

Finalement, la **conclusion** dresse un bilan du travail effectué et ouvre des voies de recherche en guise de perspectives.

Première partie

État de l'art

Chapitre 1

La complexité des biologistes

Ce travail est un travail d’informaticien. Il se veut adapté à traiter des problèmes biologiques. Il faut donc chercher à comprendre les attentes du domaine. Ce court chapitre introduit donc le domaine en faisant état des transformations profondes qui ont structuré la recherche en biologie.

Après avoir donné une définition initiale de la biologie, nous présentons l’émergence de la biologie moléculaire. Ensuite, l’évolution technique du génie génétique a conduit à l’explosion de la quantité de données. C’est alors que la bio-informatique fait son apparition. Finalement, nous constatons que ces évolutions ont posé de nouveaux problèmes qui expriment un besoin théorique. L’émergence de nouveaux paradigmes : biologie théorique, systémique ou intégrative, va alors dans ce sens.

1.1 La biologie, une science sans théorie

L’histoire naturelle des êtres vivants que les naturalistes décrivaient depuis l’antiquité est une science descriptive et classificatoire. Au siècle des lumières, l’étude des lois de la vie est progressivement complétée par la biologie, science expérimentale [Mazliak, 2006]. La biologie, du grec *bios* ($\beta\iota\omicron\varsigma$) et *logos* ($\lambda\omicron\gamma\omicron\varsigma$), est la branche des sciences qui étudie le vivant. Le terme biologie a été défini à la fin du XVIII^{ème} siècle par le naturaliste français Jean-Baptiste Pierre Antoine de Monet, Chevalier de Lamarck :

« Tout ce qui est généralement commun aux végétaux et aux animaux comme toutes les facultés qui sont propres à chacun de ces êtres sans exception, doit constituer l’unique et vaste objet d’une science particulière qui n’est pas encore fondée, qui n’a même pas de nom, et à laquelle je donnerai le nom de biologie. »

Simultanément, Gottfried Reinhold Treviranus (Biologie oder Philosophie der lebenden Natur, 1802) crée le même néologisme et en donne la définition suivante :

« La biologie envisagera les différents phénomènes et les différentes formes de vie ; les conditions et les lois qui régissent son existence, et les causes qui déterminent son activité. »

Historiquement, la biologie est considérée comme une « science molle ». En effet, par rapport à des disciplines telles que les sciences physiques, la chimie et surtout les mathématiques, la biologie est une science empirique et descriptive essentiellement basée sur l’expérience, l’expérimentation, l’observation et la constatation. L’objet biologique, le *Vivant*, est imprévisible et inconstant, et n’est pas reproductible :

- ce qui s'est produit à l'instant t ne se produira pas forcément à l'instant $t + 1$;
- deux êtres ne sont jamais exactement identiques ;
- le temps fait évoluer les êtres vivants de manière irréversible ;
- des causes identiques ne produisent donc pas nécessairement les mêmes conséquences.

Ajoutées à cela, la complexité et la diversité du vivant rendent la théorisation de la biologie extrêmement difficile. Les rares concepts théoriques présents, sont d'ailleurs souvent empruntés à la chimie ou à la physique. Par exemple, la biochimie que l'on peut considérer comme la branche « dure » de la biologie, applique les concepts et les théories de la chimie. La théorie permet notamment d'établir des modèles prédictifs pour calculer le comportement des systèmes étudiés. Ainsi les lois de la cinétique enzymatique de Henri-Michaelis-Menten permettent de décrire et d'expliquer le comportement d'une réaction biochimique. On remarque alors que les outils mathématiques sous-jacents utilisés pour établir les modèles sont souvent les mêmes depuis de nombreuses années. La section 2.1 présentera ces outils. Enfin, ajoutons que l'apport des sciences « dures » à la biologie ne s'est pas limité au partage d'outils théoriques et techniques. En plus de l'apport théorique, il y a également eu un apport humain de la physique vers la biologie.

1.2 La biologie moléculaire, l'étude de l'ordre qui engendre l'ordre

De la redécouverte des travaux de Mendel [Mendel, 1866], en 1900, sur les lois de l'hérédité, à la découverte de la structure en double hélice de l'ADN par Watson et Crick [Watson and Crick, 1953], en 1953, le $XX^{\text{ème}}$ siècle voit naître la génétique¹. Pendant cette période, les travaux de Beadle et Tatum [Beadle and Tatum, 1941] montrent que la protéine fait le lien entre le gène et le caractère observé chez l'espèce. En effet, chaque étape des voies biochimiques semble être contrôlée par un gène unique, codant l'enzyme impliquée à cette étape. Cette constatation donne naissance à l'aphorisme : *un gène \Leftrightarrow une enzyme*. Ces travaux marquent la convergence de la biochimie et de la génétique. Cette union qui sera nommée *biologie moléculaire* par William Astbury [Astbury, 1961], est fortement influencée par des physiciens qui y voient alors un intérêt scientifique majeur dans la génétique. Parmi ces physiciens d'origine, Maxime Delbrück en est alors le chef de file. Avec le microbiologiste Salvador Luria et le bactériologue Alfred Hershey, il organise le Groupe Phage, dans le but de déterminer la nature physique des gènes par l'étude d'un organisme simple : le bactériophage. Il propose la théorie selon laquelle l'être vivant serait représenté virtuellement dans les chromosomes sous la forme d'un « code ». Plus tard, un autre physicien, Erwin Schrödinger, diffusera cette théorie. Il publie en 1944 un livre très influent intitulé « *Qu'est-ce que la vie ?* » [Schrödinger, 1993] dans lequel il présente l'idée selon laquelle, en biologie, l'ordre émerge de l'ordre à l'inverse de la physique où l'ordre émerge du désordre.

La génétique (l'étude de la transmission des caractères héréditaires entre des géniteurs et leur descendance) devient alors la génomique (l'étude du génome) puisque connaissant le support matériel de la transmission des caractères, on en découvrira la structure et les propriétés dans le quart de siècle qui suivra. La période 1958 à 1969 est particulièrement faste pour la biologie moléculaire puisque six prix Nobel de physiologie ou de médecine, et

¹Le document « histoire de la biologie moléculaire » de A. Bernot et O. Alibert fournit un document historique de Mendel à Pauling, bien illustré, concis et structuré. <http://www.genoscope.cns.fr/externe/HistoireBM/>

trois prix de chimie sont attribués dans ce domaine. Cette envolée de la biologie moléculaire (ou génétique moléculaire) est bien sûr couplée aux avancées des bio-technologies.

1.3 Le génie génétique et l'explosion des données

Imaginer que la structure à une dimension de la séquence d'une molécule d'ADN représente l'information génétique d'une espèce est un concept révolutionnaire. Le séquençage de ce « code » commence en 1977 lorsque apparaissent simultanément deux techniques de séquençage, l'approche chimique de Walter Gilbert et Allan Maxam, et la méthode enzymatique de Frédérick Sanger qui prendra le pas sur la première. L'étude de la structure et des propriétés du génome subit une explosion dans les années 1980-1990. Les méthodes de production des gènes et de séquençage des génomes se sont développées. De plus, on voit apparaître une volonté internationale d'élucider complètement la séquence de génomes particuliers. Les séquences de certaines espèces ont aujourd'hui été complètement décrites : la bactérie *E.coli* (1997), le nématode *C.elegans* (1998) et la mouche drosophile (2000). En 2004, L.D. Stein annonce la fin² du séquençage du génome humain [Stein, 2004] par le projet « Human Genome Project ». La clôture du projet intervient avec deux ans d'avance, tout juste cinquante ans après la découverte de la structure de l'ADN. Cette avance traduit l'explosion du génie génétique. Le génie génétique comprend l'ensemble des techniques de la biologie moléculaire, discipline qui, comme on l'a vue, s'est structurée et définie autour de l'expression de l'information génétique et de ses régulations. Ces techniques usent de la boîte à outils enzymatique pour isoler, cloner, séquencer, manipuler les séquences et les gènes.

Le génome humain contient entre 20 000 et 25 000 gènes. Cela représente 3 milliards de paires de bases. De plus, l'ADN ne peut être séquencé que par petits bouts qui doivent être recollés. Évidemment, le séquençage des génomes conduit à une explosion des données.

1.4 Bio-informatique

La quantité de données biologiques ainsi générée rend le traitement automatisé de ces données nécessaire à leur analyse [Reichhardt, 1999]. Les méthodes automatiques requises pour stocker, traiter et analyser cette information biologique sont historiquement regroupées sous l'appellation *bioinformatics*. Pour les anglo-saxons, le terme de *bioinformatics*, né dans les années 90, désigne la discipline spécifiquement consacrée à l'étude des séquences protéiques et des structures des réseaux d'interactions moléculaires. Nous utilisons, ici, le terme *bio-informatique* de manière réductrice, en limitant son sens à celui de son faux-ami anglo-saxon. Nous pouvons faire un tour d'horizon de la discipline en distinguant différentes tâches qu'elle accomplit.

1.4.0.1 Gestion et Organisation des données

De plus en plus de séquences de génomes, de transcriptomes ou de protéomes sont devenues disponibles. Afin de traiter ces informations, les bio-informaticiens ont besoin de stocker, d'échanger, de récupérer les données. La première difficulté est donc d'organiser cette énorme masse d'information et de la rendre disponible à l'ensemble de la communauté des chercheurs.

²Une fin relative puisqu'il reste à séquencer 20% de l'ADN qui repose sur des zones dites *pauvres en gènes* qui sont impliquées dans les processus de réplifications chromosomales et de *maintenance*. Le problème est que les techniques actuelles (i.e. enzymatiques) ne permettent pas de séquencer ces zones. Il faut donc s'attendre à d'autres annonces de la fin du séquençage du génome humain d'ici 2010.

Pour cela, il existe un grand nombre de bases de données biologiques qui peuvent être d'intérêt biochimique, génétique, pharmaceutique. En janvier 2006, le journal *Nucleic Acid Research* recensait plus de 800 banques de données [Galperin, 2006].

La principale mission des banques est de rendre publique les données issues de fonds publics, donc collectives. Pour cela, un processus de double publication s'est imposé depuis 1988 : les chercheurs doivent ainsi déposer leurs séquences aux banques avant de soumettre l'article associé aux revues scientifiques.

Le Centre de Ressources Infobiogen³ constitue par exemple un outil de référence en France pour accéder aux données biologiques relatives à la génomique.

Via le système d'interrogation SRS, le serveur Infobiogen maintient et met à la disposition de ses utilisateurs un ensemble de bases de données génétiques régulièrement actualisées, comprenant :

- les grandes banques généralistes internationales de séquences nucléiques (EMBL [Kanz et al., 2005] et Genbank [Benson et al., 2006]) et protéiques (Uni-Prot [Bairoch et al., 2006]).
- des banques spécialistes : banques de motifs, de données structurales ou de données cartographiques.

Les utilisateurs ont accès localement, ou sur les serveurs publics, à deux types de logiciels pour utiliser ces bases. Les premiers logiciels sont des systèmes d'interrogation dédiés qui sont programmés exclusivement pour la manipulation de séquences biologiques (SRS). Les seconds sont des programmes établis à l'aide de Systèmes de Gestion de Bases de Données (Oracle, Sybase, 4D...) qui utilisent un langage de requêtes standard et un format de stockage des données indépendant de la nature de l'information contenue dans les bases. Nous pouvons citer un article de 1998 [Kastin and Wexler, 1998] qui associe le terme *bioinformatics* aux moteurs de recherche classiques de la toile.

1.4.0.2 Comparer, analyser, explorer les informations

Le champ disciplinaire de la génomique a constitué le moteur principal de la bio-informatique. En raison de la taille et de la complexité des génomes, leur étude passe par trois étapes principales :

- Le séquençage : détermination de l'ordre d'apparition des nucléotides qui composent une molécule d'ADN.
- La cartographie : l'établissement de cartes (physiques ou de liaison), qui permettent de se repérer le long des chromosomes à l'aide de marqueurs.
- L'analyse des séquences : prédiction de structure de gène, de structure 2D ou 3D de protéine, recherche de similarité, phylogénie.

Sans entrer dans les détails, nous pouvons essayer de recenser les techniques informatiques mises en jeux dans ces tâches. L'étape du séquençage ne fait pas réellement appel à la bio-informatique mais plutôt aux bio-technologies (ex : séquenceurs automatiques). Évidemment, les techniques de gestion de bases de données sont largement utilisées, et sont un préalable à l'utilisation des résultats du séquençage.

³Le Centre de Ressources INFOBIOGEN (CRI) a été créé le 10 juin 1999 par le Ministère de l'Éducation Nationale, de la Recherche et de la Technologie, et l'Université d'Evry Val d'Essonne. Cette nouvelle structure remplace le Groupement d'Intérêt Scientifique (GIS) INFOBIOGEN en activité depuis 1995. www.infobiogen.fr

Ensuite, lorsque l'on a séquencé une portion d'ADN, on cherche à savoir si quelqu'un a déjà rencontré ce type de séquence. Ainsi, la cartographie et l'analyse de séquences vont très largement mettre en œuvre des techniques d'algorithme du texte, parmi lesquelles se trouve la recherche de similarités qui demeure un grand classique de l'intelligence artificielle computationnelle. Elle a été appliquée pour la première fois à l'étude des séquences protéiques grâce à l'algorithme d'alignement global de Needleman et Wunsch en 1970 [Needleman and Wunsch, 1970]. Depuis, on distingue trois algorithmes principaux qui recherchent les similarités locales :

- Smith Waterman [Smith and Waterman, 1981] : s'inspire de Needleman et Wunsch, utilise une approche par programmation dynamique [Bellman, 1957].
- Fasta [Pearson and Lipman, 1988] : basé sur une heuristique (donc plus rapide, mais moins sensible), est un logiciel de comparaison de séquences avec insertion/délétion.
- Blast [Altschul et al., 1990] : également basé sur une heuristique, Blast utilise des méthodes statistiques. C'est l'algorithme le plus rapide des trois, mais aussi le moins sensible.

La recherche de similitudes est souvent utilisée par la classification. La tâche qui consiste à reconstruire des arbres phylogéniques de séquences, met à contribution tout l'arsenal de la classification [Lecointre and Le Guyader, 2001]. On trouve des algorithmes basés sur la distance, la parcimonie, ou le maximum de vraisemblance. Les algorithmes de prédiction de fonctions de protéines font en quelque sorte aussi appel à la recherche de similarité, avec d'un côté le criblage virtuel et l'amarrage par similarité sur des modèles en trois dimensions, et de l'autre la phylogénie moléculaire.

Depuis des années, les bio-informaticiens se sont familiarisés progressivement avec des concepts aussi divers que le recuit simulé, les chaînes de Markov, et les statistiques bayésiennes. Nous pouvons retenir (en anticipant sur le chapitre suivant) que la bio-informatique fait plutôt appel à un aspect computationnel de l'informatique. La bio-informatique est le traitement automatique de l'information biologique, sous forme de données. La base de données (données primaires) est donc la matière première à partir de laquelle la bio-informatique va produire d'autres données (données secondaires) et construire de nouvelles connaissances.

L'histoire de la bio-informatique est liée à celle de la biologie. Elle a usé d'une démarche analytique commandée par la biologie moléculaire : isoler la séquence pour mieux l'analyser et en étudier la richesse syntaxique ; ce qui ne va pas sans poser de gros problèmes conceptuels. Ainsi, depuis quelques années, on observe une évolution de la pensée en biologie pour résoudre ces problèmes.

1.5 Changement de méthode : biologie systémique, théorique ou intégrative

ù La biologie moléculaire et cellulaire, la génétique et la biochimie ont mis à jour une grande quantité de faits biologiques tels que les séquences génomiques et les propriétés de certaines protéines. Il est apparu que la connaissance exhaustive de la structure des êtres vivants est insuffisante pour leur interprétation et pour répondre à la question posée par E.Schrodinger : « Qu'est-ce que la vie ? ». Un changement dans la pensée biologique a donc dû s'opérer. L'histoire de ce changement de méthode est due à plusieurs raisons.

La principale raison, que nous venons de citer, est la demande de nouveaux paradigmes posée par l'héritage du projet génome humain. C'est pour cela et parce que son impact dans l'usage de l'informatique pour la biologie est déterminant que nous avons décrit cette

histoire. Le livre « ni dieu ni gène » [Kupiec and Sonigo, 2000] illustre bien les limites de la vision mécaniste de la biologie moléculaire classique.

Ensuite, l'expérience fournie par les autres domaines scientifiques, ont montré le pas pour intégrer la complexité dans la démarche scientifique. La révolution de la physique de la matière a conduit à un changement de paradigme⁴. Dans les années 1920, les physiciens (Bohr, de Broglie, Schrödinger, Pauli, Heisenberg, Dirac...) ont alors du changer leur vision de la nature de la matière et sa relation avec l'esprit humain [Capra, 1975].

Enfin la philosophie de la biologie a connu différents mouvements qui ont permis de répondre aux questions posées : l'écologie radicale, le vitalisme, la biologie organisciste, la théorie de la gestalt sont autant de tendances qui ont cherché à décrire la nature du vivant en s'opposant au réductionnisme. L'ouvrage « la toile de la vie »[Capra, 2003] propose un résumé remarquablement accessible de ce cheminement.

Finalement on peut dénombrer trois dénominations pour exprimer la vision émergente du vingtième siècle : la biologie systémique ; la biologie intégrative ; la biologie théorique. « Biologie Systémique » est le terme que nous adopterons le plus facilement car c'est celui utilisé par la communauté informatique. Il ressemble de plus au terme anglo saxon *systems biology* qui rassemble les travaux tels ceux que nous exposerons dans ce document. Nous pouvons comprendre la biologie systémique comme l'application de l'approche systémique à la biologie.

L'Association Française de Science des Systèmes (AFSCET) fournit un document⁵ définissant de manière consensuelle l'approche systémique qui est :

- plus dominée par une logique ternaire ou conjonctive (qui relie) que par une logique binaire ou disjonctive (qui sépare),
- plus centrée sur le but à atteindre (finalité) que sur la recherche des causes (causalité),
- plus relationnelle et globale qu'analytique,
- plus orientée par le présent-futur (prospective) que par le passé-présent (déterminisme),
- plus ouverte sur la diversité des réalités et la pluralité des solutions que sur la quête de certitudes et de réponses « universelles » (the one best way),
- moins réductrice enfin car accueillante à l'émergence de la nouveauté et à l'invention.

La systémique est indissociable de la modélisation, et de ce fait, elle a très souvent recours à l'informatique. Elle fait appel aux concepts de système, d'interaction, de rétroaction, de régulation, d'organisation, de vision globale, d'évolution, *etc* [Querrec, 2005]. Nous définirons ces concepts dans le contexte de cette thèse, lorsque nous y ferons appel.

Le terme « biologie intégrative » a lui plus d'affinité avec les « vrais » biologistes. Il véhicule la même volonté de considérer les relations entre les composants biologiques et d'étudier des systèmes à différents niveaux d'organisation. On retrouve le plus souvent ce terme dans des travaux concrets de biologistes ne faisant pas forcément appel à l'informatique et ne s'embarrassant pas directement des concepts de la systémique même s'ils sont sous-jacents. La notion d'intégration peut alors prendre plusieurs sens :

- L'intégration d'une masse importante de données ou de connaissances pour donner du sens à l'information disponible.
- L'intégration « verticale » qui permet le passage d'un niveau d'organisation élémentaire

⁴Nous pouvons citer la définition du paradigme de T. Kuhn [Kuhn, 1972] : « une constellation de réussites -concepts, valeurs, techniques, etc.- que partage une communauté scientifique et qu'elle utilise pour définir la validité des problèmes et des solutions ». Lorsqu'une révolution a lieu dans un domaine scientifique, on assiste à un changement de paradigme.

⁵www.afscet.asso.fr/SystemicApproach.pdf

à un niveau d'organisation plus complexe.

- L'intégration « transversale » qui permet de transposer des connaissances acquises sur une espèce à d'autres espèces.

Le terme plus général, « biologie théorique » recoupe les deux précédents. Il semble s'adresser le plus souvent à des travaux plutôt philosophiques.

Ainsi, le nouveau paradigme conduit à considérer les cellules, les tissus, les organes, les organismes et les écosystèmes comme des systèmes de composants en interaction qui doivent être appréhendés dans leur complexité. Les différents domaines de la biologie ont été marqués par ce changement. En physiologie, un glissement s'opère de la fonction (mécaniste) vers l'organisation (pensée systémique) en physiologie. La génétique était devenue la génomique (voir section 1.2) elle redevient la génétique pour considérer les biomolécules et autres objets biologiques non plus isolément mais dans leur environnement, en interaction les uns avec les autres. Elle tente de recombinaison le tout à partir de ses éléments en tenant compte du jeu de leurs interdépendances et de leur évolution dans l'espace et dans le temps. Ceci se traduit par le développement des sciences en « omiques » : génomique, transcriptomique, protéomique, métabolomique, physiomique [Auffray et al., 2003b]. La complexité intrinsèque des systèmes biologiques traitée par la méthode systémique rend nécessaire la combinaison d'approches expérimentales et informatiques. L'outil informatique doit donc être pensé pour la complexité. Le chapitre suivant présentera cet aspect.

1.5.0.3 Élargir le champ d'application : les scopes

En guise de clin d'œil, nous pouvons citer le fleurissement des « scopes » pour illustrer les orientations de la biologie actuelle. En 1975, Joël De Rosnay introduit en France la vision systémique en proposant un outil symbolique : le « microscope » [De Rosnay, 1975]. Le microscope doit être à la méthode systémique ce que le microscope est à la méthode réductionniste. Il doit permettre d'appréhender l'infiniment complexe. Cette idée de microscope ne demande qu'à être concrétisée. Ainsi différents programmes de recherche ont vu le jour.

Le « Géoscope », Centre National de Séquençage, est le premier « grand équipement » français en biologie. La mission du Géoscope est de produire et d'interpréter de grands volumes de données et de séquences de haute qualité.

Le consortium international « Systemoscope » [Auffray et al., 2003a] est un réseau international pour développer un programme de recherche et de formation trans-disciplinaire en biologie systémique, fondé sur l'expérience en mathématique, informatique, physique, biologie et médecine de ses participants. L'intérêt majeur d'un tel projet est la mise en œuvre de recherches interdisciplinaires. Car, bien que l'interdisciplinarité soit le maître mot des observateurs de la recherche liée à la biologie, elle reste en pratique extrêmement difficile à établir.

Le Virtuoscope [Fuchs et al., 2006], propose d'utiliser la réalité virtuelle pour « implémenter » une sorte de microscope. Ce dernier nous intéresse particulièrement puisque ce travail s'intègre dans ce projet du Centre Européen de Réalité Virtuelle (CERV). Le chapitre 3 décrira ce projet.

Remarquons que, dans ces trois « scopes », l'usage de l'informatique est essentiel à leur réalisation.

Chapitre 2

Complexité biologique et informatique

La biologie a intégré la complexité par le biais de la biologie systémique. L'informatique comme outil de modélisation aidant les biologistes dans cette démarche, doit également prendre en compte cette complexité. Ce qui nous intéresse ici c'est d'étudier comment l'informatique peut servir à manipuler la complexité du vivant. Nous constatons que deux types d'analyse existent dans ce contexte : l'analyse *a priori* et l'explication *a posteriori*.

Dans ce chapitre, nous nous arrêtons en premier lieu sur les domaines connexes des mathématiques pour l'étude de la complexité. Nous décrivons ensuite les méthodes informatiques formelles pour l'analyse des systèmes vivants. Nous précisons également la notion de calcul *in silico* que nous associons en majeure partie à la simulation (explication *a posteriori*). Finalement, nous nous intéressons aux méthodes de simulation.

2.1 Des mathématiques pour l'étude de la complexité

L'histoire de l'informatique est liée à celle des mathématiques. D'un certain point de vue l'informatique peut être vue comme une branche particulière des mathématiques. Évidemment, les mathématiciens n'ont pas attendu l'ère de l'informatique pour traiter la complexité. Cependant, l'avènement de l'ère informatique a permis l'application de certaines méthodes mathématiques. À l'inverse, lorsque nous présentons les outils informatiques, les concepts mathématiques sont souvent présents. Ensuite précisons que nous devons être prudents avec les notions de complexité. La complexité des mathématiques, de la physique ou de la biologie sont parfois très différentes. Ainsi l'objectif de cette section est de présenter certaines notions qui sont utiles à la compréhension de l'usage de l'informatique pour la formalisation, la modélisation et la simulation des systèmes biologiques.

Cette section se divise alors en deux parties. La première recense les méthodes numériques de résolution de systèmes d'équations différentielles. La seconde, présente certains concepts d'ordre général qui sont parfois sous-jacents aux travaux exposés par la suite.

2.1.1 Le calcul infinitésimal

Les sciences auxquelles la biologie emprunte certain éléments théoriques, possèdent l'outil quasi exclusif, depuis plus de deux cents ans, des calculs et simulations de phénomènes phy-

siques, les équations différentielles. Elles sont issues de l'invention simultanée de Leibniz et de Newton, le calcul infinitésimal. Le calcul infinitésimal ou différentiel se base sur l'idée que le temps est une variable continue. Il permet d'étudier efficacement les systèmes linéaires. Historiquement, les physiciens ont cherché à décrire les phénomènes par des systèmes d'équations différentielles. Les résolutions analytiques des modèles mathématiques permettent de trouver les équations qui prédisent le comportement des systèmes étudiés. Malheureusement, les méthodes analytiques sont rarement envisageables, du fait de la non-linéarité des équations. Pour surmonter ce problème, les modélisateurs ont cherché à linéariser les équations décrivant les phénomènes, quitte à les dénaturer. Depuis, les possibilités de calcul offertes par l'informatique ont donné naissance à un arsenal de méthodes numériques, pour traiter les systèmes d'équations différentielles, en conservant leur non-linéarité.

En biologie, lorsqu'elle se place à un niveau macroscopique¹, l'étude de la cinétique biochimique est la principale consommatrice de calculs numériques à base d'équations différentielles. Les phénomènes qu'elle étudie sont le plus souvent des réseaux de réactions chimiques faisant intervenir des réactions enzymatiques. En général, les milieux dans lesquels les réactions se réalisent sont considérés homogènes. Il n'y a alors pas de dimension spatiale ; ce qui conduit à étudier des systèmes d'équations différentielles ordinaires (EDO). Les réactions enzymatiques se modélisent alors par les lois cinétiques du type Michaelis-Menten [Michaelis and Menten, 1913] qui ne sont pas linéaires. Dans ce cas la solution analytique ne peut être trouvée et le chercheur doit faire appel à la simulation pour observer le comportement du système qu'il étudie. Classiquement les systèmes d'EDO s'expriment sous la forme d'un problème de Cauchy : on cherche à calculer une solution à partir de l'état initial du système. La communauté de la simulation informatique a donc adapté les méthodes de résolution numérique aux travers de logiciels spécifiques.

Le problème incontournable des calculs numériques est qu'un microprocesseur n'effectue les instructions que de manière séquentielle. Le temps doit donc être discrétisé. Cela entraîne nécessairement une erreur numérique de la méthode qui approxime les valeurs de la solution.

L'idée est alors d'utiliser des méthodes numériques ayant un ordre élevé, celui-ci reflétant la précision de la méthode (plus il est élevé, plus la méthode est précise).

Le second critère qui intervient est le temps de calcul. En effet, les méthodes qui minimisent le plus l'erreur numérique ont tendance à faire accroître considérablement le temps de calcul. L'art de la simulation consiste donc à choisir la méthode qui produira une erreur acceptable pour un temps de calcul raisonnable.

Les techniques numériques couramment utilisées sont les suivantes :

- **Euler explicite** : C'est la méthode de base de résolution numérique d'EDO. A chaque pas, on calcule la dérivée de la solution à l'instant t , puis on intègre cette dernière que l'on considère constante jusqu'à $t + 1$.
- **Contrôle du pas** : Afin de réduire l'erreur numérique de la méthode d'Euler, il est possible de faire varier le pas d'intégration en fonction de l'erreur commise sur celui-ci. Toutes les méthodes modernes utilisent le contrôle du pas pour réduire considérablement le temps de calcul.
- **Euler implicite** : Les problèmes de la cinétique biochimique sont souvent des problèmes

¹À ce niveau on considère les réactions chimiques faisant intervenir des concentrations de molécules et non les molécules prises individuellement (niveau microscopique)

« raides ». Dans ces cas particuliers, il devient très coûteux de conserver la A-stabilité de la méthode numérique. C'est-à-dire que la solution trouvée oscille autour de la solution exacte. La méthode d'Euler implicite permet de conserver la A-stabilité. Le principe est de calculer la dérivée à l'instant $t + 1$ et de l'appliquer à l'instant t . Cette technique nécessite la résolution d'un système non linéaire pour calculer la solution à chaque pas de simulation : son coût est donc très supérieur à Euler explicite.

- **Runge-Kutta :** Tout comme la méthode d'Euler, les méthodes de Runge-Kutta sont des méthodes à un pas. Il en existe de plusieurs sortes. Le principe est d'utiliser des valeurs intermédiaires dans le pas d'intégration afin d'augmenter leur précision numérique. On trouve par exemple la méthode du point milieu ou trapézoïdale, la plus célèbre étant la méthode de Runge-Kutta d'ordre 4, qui offre un excellent rapport précision/temps de calcul. Il existe également des méthodes implicites comme la méthode de Rosenbrock. On trouve aussi des techniques à pas adaptatif, dont les méthodes emboîtées explicites (Runge-Kutta-Fehlberg d'ordre 4(5)) et implicites (Dormand-Prince 5(4)).
- **Pas multiples :** Le principe des méthodes à pas multiples est d'utiliser les valeurs aux points précédemment calculés afin d'augmenter l'ordre et la précision. L'avantage est qu'elles demandent moins d'évaluations de fonction. En contre partie, elles nécessitent une plus grande capacité de stockage. Les méthodes les plus courantes sont les méthodes d'Adams pour les systèmes non raides, et les méthodes BDF (*Backward Differentiation Formulas*) pour les problèmes raides. Finalement, il existe des implémentations telles LSODE, LSODA et CVODE qui combinent les deux familles de méthodes, suivant que le système d'EDO est raide ou non.

Pour une étude détaillée et approfondie de la résolution numérique des EDO dans le cas général voir [Ascher and Petzold, 1998] et [Hairer et al., 1993; Hairer and Wanner, 1996]. La thèse de Sébastien Kerdélo [Kerdélo, 2006] fournit un état de l'art détaillé et récent des méthodes numériques de résolution d'équations différentielles ainsi que les simulateurs qui les mettent en œuvre. Les tableaux 2.1 et 2.2 montrent les principaux outils résolvants des systèmes d'EDO dans le cadre de la cinétique biochimique.

Le nombre des simulateurs d'EDO dédiés à la cinétique biochimique s'est beaucoup accru ces dernières années. On peut remarquer que ces logiciels ont tendance à se fédérer autour de la norme émergente SBML (voir section 2.4.2). Aujourd'hui, l'intérêt semble se porter sur l'ajout de dimensions spatiales dans les systèmes de réactions biologiques. Il faut donc traiter des systèmes d'équations aux dérivées partielles (EDP). Actuellement, seul VirtualCell [Slepchenko et al., 2002] permet de traiter les EDP dans un contexte dédié à la biologie.

Cette présentation de l'outil clef qu'est le calcul infinitésimal et des méthodes numériques associées est maintenant achevée. Bien entendu, ces méthodes de résolution ne sont pas propres à l'étude de systèmes complexes, mais elles permettent de produire des résultats analysables.

2.1.2 Les systèmes complexes

2.1.2.1 Les systèmes dynamiques

Un système dynamique évolue au cours du temps de manière causale. L'état d'un système est défini par des variables dynamiques qui représentent les grandeurs physiques (concentrations, positions, vitesses...) qui déterminent l'état instantané du système et qui ne sont pas

	Méthodes à un pas								Contrôle du pas	Détection dynamique de la raideur	Référence bibliographique
	Explicites				Implicites						
	Point milieu	Trapézoïdale	Runge-Kutta 4 classique	Runge-Kutta Fehlberg	Dormand-Prince	Point milieu	Trapézoïdale	Rosenbrock			
SIMFIT				×						×	[Holzhütter and Colosimo, 1990]
SCAMP				×						×	[Sauro and Fell, 1991; Sauro, 1993]
MIST							×			×	[Ehlde and Zacchi, 1995]
VIRTUAL CELL	×	×	×	×						×	[Schaff et al., 1997, 2000, 2001; Loew and Schaff, 2001; Moraru et al., 2002; Slepchenko et al., 2003]
NEURON						×	×				[Hines and Carnevale, 1997; Lytton and Hines, 2005]
GENESIS						×	×				[Bower and Beeman, 1998]
E-CELL			×		×					×	[Tomita et al., 1999; Takahashi et al., 2003, 2004]
A-CELL			×								[Ichikawa, 2001]
XPP/XPPAUT	×	×	×		×			×		×	[Ermentrout, 2002]
DYNETICA			×							×	[You et al., 2003]
CELLWARE			×						×	×	[Dhar et al., 2004]
DIZZY				×	×					×	[Ramsey et al., 2005]

TAB. 2.1: Récapitulatif des méthodes à un pas [Kerdélo, 2006].

	Méthodes à pas multiples						Contrôle du pas	Détection dynamique de la raideur	Référence bibliographique
	Explicites		Implicites		Mixtes				
	Adams-Bashforth	Prédiction-Correction	Adams-Moulton	BDF	LSODE	LSODA			
SCAMP						×		×	[Sauro and Fell, 1991; Sauro, 1993]
GEPASI						×		×	[Mendes, 1993, 1997; Mendes and Kell, 1998, 2001]
DYNAFIT					×			×	[Kuzmic, 1996]
VIRTUAL CELL			×			×		×	[Schaff et al., 1997, 2000, 2001; Loew and Schaff, 2001; Moraru et al., 2002; Slepchenko et al., 2003]
NEURON							×	×	[Hines and Carnevale, 1997; Lytton and Hines, 2005]
GENESIS	×								[Bower and Beeman, 1998]
DBSOLVE					×			×	[Goryanin et al., 1999]
JARNAC						×	×	×	[Sauro, 2000]
XPP/XPPAUT		×		×			×	×	[Ermentrout, 2002]

TAB. 2.2: Récapitulatif des méthodes à pas multiples [Kerdélo, 2006].

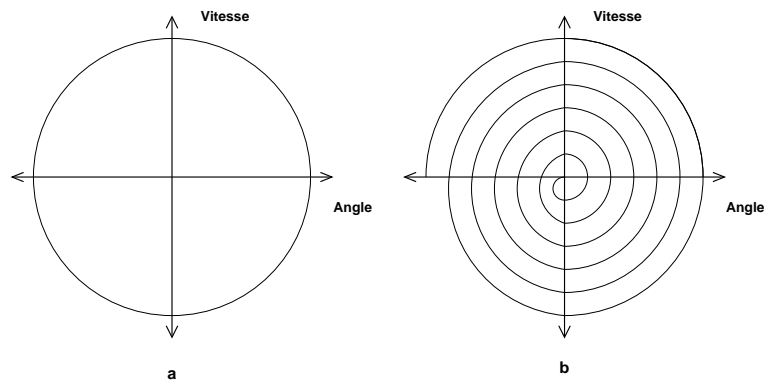


FIG. 2.1: La figure a représente un attracteur périodique du mouvement d'un pendule. La figure b représente l'attracteur ponctuel d'un pendule soumis aux frottements ([Arnold, 1981]).

constantes *a priori*. Deux types de systèmes sont considérés : les systèmes à temps discret [Robert, 1995] et ceux à temps continu [Petrovsky, 1988].

L'étude des systèmes dynamiques a réellement émergé grâce à l'essor de l'informatique qui rend possible la résolution d'équations complexes et la visualisation des résultats sous forme graphique. En effet, les outils de résolution d'équations différentielles présentés en section 2.1.1, permettent parfois de trouver des solutions numériques aux équations non linéaires. La théorie des systèmes dynamiques permet d'analyser ces résultats. Ceci favorisant la découverte de *patterns* qualitatifs de comportement des systèmes complexes, représentant un niveau d'ordre sous-jacent au chaos apparent.

2.1.2.2 Systèmes non-linéaires

Les systèmes vivants étant organisés en réseaux possédant de nombreuses boucles de rétroaction, présentent des comportements fondamentalement non linéaires. La théorie des systèmes dynamiques fournit un des premiers outils permettant d'étudier les équations non linéaires de la complexité [Petrovsky, 1988]. Les systèmes non linéaires font apparaître les trois propriétés suivantes :

- Une équation déterministe simple peut produire une grande variété de comportements ; par ailleurs, un comportement complexe et chaotique peut donner naissance à un comportement ordonné ;
- Il est parfois impossible de faire des prédictions exactes malgré le déterminisme des équations envisagées ;
- De petits changements dans les conditions initiales peuvent avoir des effets spectaculaires sur les prédictions.

En utilisant les notions d'attracteurs (voir figure 2.1) et de points de bifurcation, les mathématiciens offrent un regard différent sur le comportement des systèmes. Un attracteur peut être vu comme la représentation du comportement d'un système dans l'espace des phases (par exemple, l'angle et la vitesse pour le pendule). Outre les attracteurs ponctuels et périodiques des systèmes simples, on trouve la notion d'« attracteur étrange » pour les systèmes chaotiques.

2.1.2.3 Les bifurcations

En 1963, le météorologue Edward N. Lorenz propose un système non linéaire à seulement trois variables qui permet d'obtenir une dynamique très complexe en dépit de la simplicité

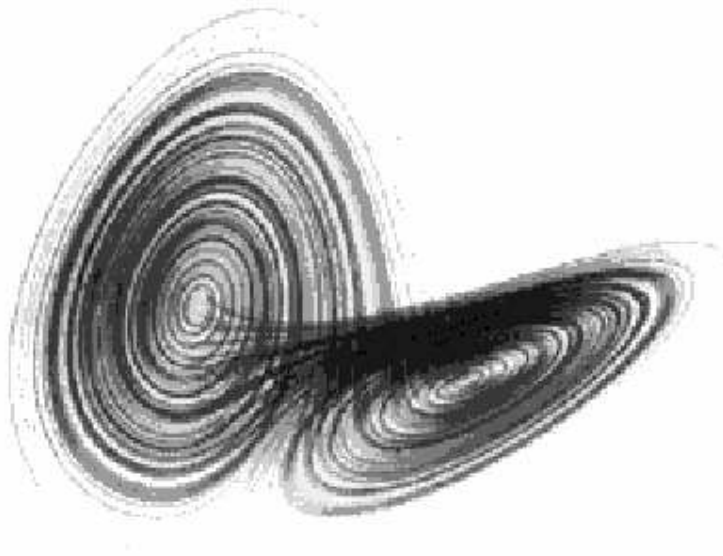


FIG. 2.2: Cette figure représente l'attracteur étrange de Lorenz (1963).

formelle du système. Il montre par ce moyen que la complexité est intrinsèque au système. La figure 2.2 montre l'attracteur de ce système. En règle générale, la connaissance des attracteurs permet de savoir partiellement (c'est-à-dire ici, au moins statistiquement) ce qui va émerger du chaos, alors que la connaissance des éléments individuels du système chaotique n'y aide en rien.

Alors que le système semble désordonné (aucun point n'apparaît deux fois), l'attracteur semble montrer une forme d'ordre. Ici, nous remarquons que le système de Lorenz semble osciller autour de deux points. L'attracteur nous donne ainsi des informations qualitatives sur le système qui peut bifurquer dans deux états différents. Il est impossible de prévoir à quel point de l'espace des phases, le système passera à un instant donné, tant le système est sensible aux conditions initiales. C'est un système chaotique.

Pour analyser les systèmes soumis à des bifurcations [Wiggins, 1990] il est possible d'utiliser les diagrammes de bifurcations. Un diagramme de bifurcations représente les états atteignables par un système en fonction de la valeur d'un paramètre. Cela permet de déterminer les intervalles de valeurs d'un paramètre pour lesquels un système pourra atteindre certains bassins d'attraction. La figure 2.3 reprend un exemple classique de diagramme de bifurcations.

Les diagrammes de bifurcations ne sont bien entendu pas les seuls outils pour étudier les systèmes dynamiques. Surtout, ils permettent l'analyse *a posteriori* des résultats. Intéressons nous maintenant à d'autres méthodes, ancrées dans la logique, formalisant les systèmes pour leur analyse *a priori*.

2.2 Les formalismes « bio-logiques »

Formaliser un système lors de son analyse, c'est donner une forme logique à ses éléments en faisant abstraction de la matière ou du contenu du système ; ceci pour permettre la

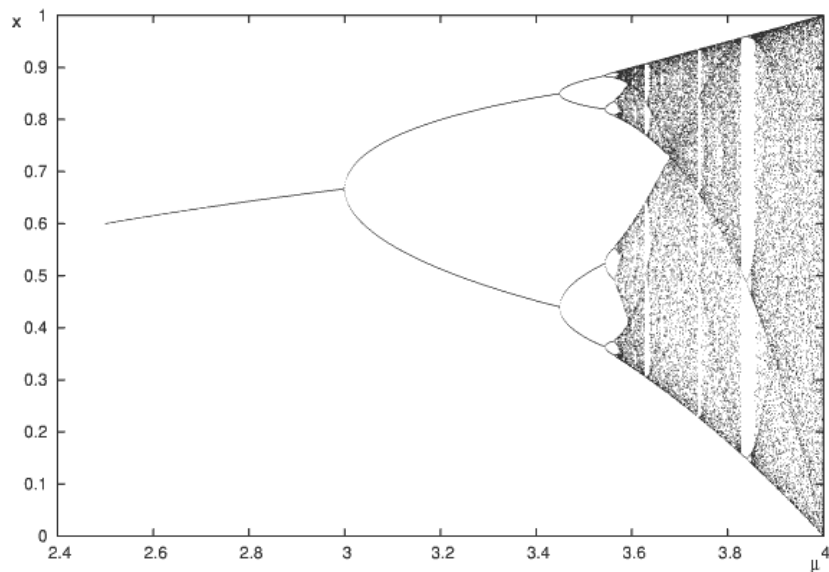


FIG. 2.3: Diagramme de bifurcations de la suite logistique : $x_{n+1} = \mu \cdot x_n \cdot (1 - x_n)$ et $x_0 \in [0; 1]$. Sont représentés sur le diagramme les valeurs des termes x_n pour n suffisamment grand, en fonction de la valeur de μ . On remarque que suivant les zones de valeurs dans lesquelles se trouve μ , la suite peut prendre 1, 2, 4, 8, etc... valeurs, jusqu'à devenir chaotique.

découverte de propriétés ou le traitement automatique de l'analyse, là où l'analyse intuitive est complètement inenvisageable. La formalisation passe par l'utilisation d'un langage formel composé d'une syntaxe et d'une sémantique. Une méthode formelle permet de raisonner rigoureusement et *a priori* sur un système, en se basant sur des raisonnements de logique mathématique. Suite aux découvertes de la biologie moléculaire, la communauté scientifique a exprimé le besoin de formaliser les systèmes vivants.

Dans cette quête, les informaticiens, à l'instar des mathématiciens, ont naturellement cherché à réinvestir leur patrimoine théorique dans l'analyse des réseaux génétiques, et plus généralement, dans celle des réseaux d'interactions moléculaires. Ainsi, ces dernières années sont les témoins d'une explosion du nombre d'études se lançant dans de nouvelles pistes réutilisant, modifiant et adaptant des formalismes déjà bien connus. Toutes les techniques classiques des sciences de l'information semblent être mises à contribution, si bien qu'il est difficile d'y voir clair dans ce domaine émergent. De plus, le contexte interdisciplinaire et donc la variété dans les points de vues altèrent la lisibilité de ces travaux. Parfois c'est l'objectif biologique d'une étude qui prime, parfois c'est l'aspect technique informatique, parfois c'est l'intérêt systémique. Le but de chacun est de proposer « le » langage du vivant et de le valider sur des exemples bien connus.

2.2.1 La complexité des réseaux

Des bases de données telle KEGG [Kanehisa et al., 2006] fournissent des informations sur les interactions entre gènes, ARN et protéines. Les enchevêtrements de ces interactions constituent des réseaux.

Souvent, les connaissances sur ces systèmes sont plutôt qualitatives. Il arrive ainsi que

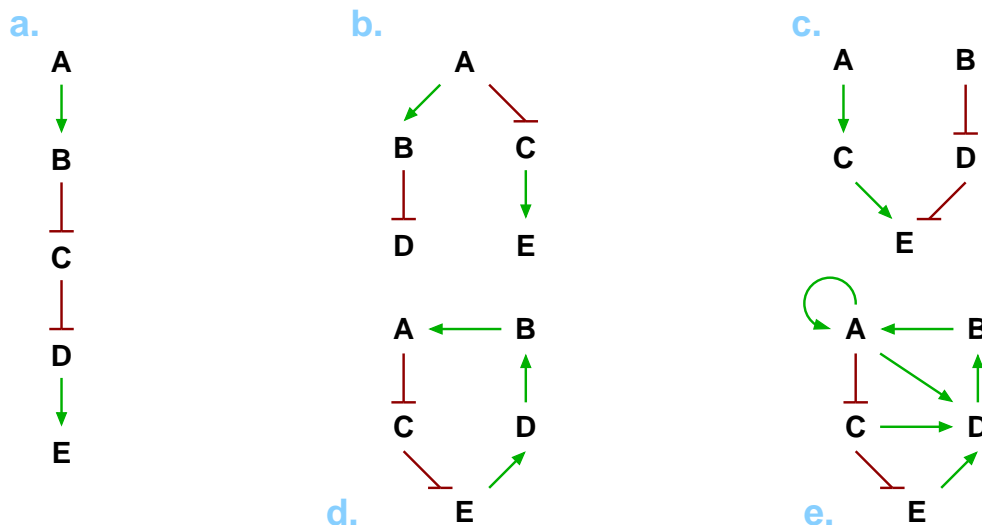


FIG. 2.4: Quelques exemples illustrant le problème de la complexité dans les réseaux d'interactions génétiques (repris de [Thieffry and De Jong, 2002]). Les sommets font référence aux gènes, les flèches représentent les interactions globales entre les gènes (inhibition en rouge, activation en vert). Dans les cas *a*, *b* et *c* il est facile de prédire la répercussion de la suppression ou de la perte de fonction d'un gène. Le cas *d* illustre une cascade où l'expression d'un gène dépend de l'expression d'un autre gène, en influence un autre et dépend indirectement d'elle même. Le graphe *e* montre ce qu'il se passe lorsqu'on imbrique plusieurs circuits ; l'analyse intuitive devient rapidement impossible.

l'interaction entre deux molécules soit recensée, mais que sa cinétique ne soit pas précisément décrite. Et, quand bien même elle le serait, nous pouvons nous demander si cette information serait pertinente dans de tels systèmes. En effet, d'un point de vue systémique, ce sont jusqu'à 10.000 protéines qui peuvent être impliquées dans un processus de régulation de la cellule humaine. Décrire un de ces grands réseaux complexes par des équations différentielles n'est donc pas forcément le plus adapté. La taille d'un modèle décrivant précisément la cinétique d'un réseau d'interactions entre des milliers de types moléculaires surpasserait très largement les capacités de traitement des ordinateurs.

Finalement, il paraît préférable de s'intéresser aux liens entre la structure des réseaux et les fonctions qu'ils réalisent. Dans ce cas, les informaticiens ont plutôt tendance à vouloir utiliser les logiques ou la théorie des graphes qui se prêtent mieux à l'étude de systèmes qualitatifs ; la complexité inhérente aux réseaux (figure 2.4) même de petites tailles, étant plus facilement appréhendable par des outils qualitatifs.

En 2002, Thieffry et De Jong [Thieffry and De Jong, 2002] classaient les méthodes de traitement des réseaux génétiques en quatre catégories principales faisant respectivement appel à la théorie des graphes, à une formalisation discrète ou logique, aux systèmes d'équations différentielles, ou encore à des équations stochastiques (voir 2.3.5.3). Aujourd'hui, la communauté s'intéresse à des formalismes hybrides permettant de combiner les différentes approches. Nous pouvons citer les méthodes suivantes :

- les algèbres de processus ;
- les réseaux de Petri hybrides et fonctionnels ;
- les logiques temporelles.

Ne pouvant être exhaustif, nous décrivons ici trois méthodes formelles pour donner un aperçu des travaux que nous regroupons sous l'appellation « bio-logiques ».

2.2.2 Les algèbres de processus

La concordance entre le comportement complexe et collaboratif des réseaux moléculaires et celui des programmes communicants de l'informatique distribuée, a conduit des informaticiens impliqués dans des programmes de recherche interdisciplinaire à adapter l'algèbre de processus à l'étude biologique.

L'algèbre de processus est une algèbre. Autrement dit, nous avons affaire à un système formel comportant un alphabet et un procédé de production de mots. À partir d'axiomes et de règles d'inférences, il devient possible de faire de la déduction, de la négation de proposition, du raisonnement par récurrence, ou de prouver l'exactitude ou le caractère erroné d'un théorème.

La syntaxe de l'algèbre de processus permet de représenter des processus, des actions et des opérateurs. La sémantique opérationnelle d'une algèbre de processus associe un modèle à un terme de cette algèbre, sous la forme d'un automate à états finis. Plusieurs formalismes permettent d'enrichir l'algèbre élémentaire des processus. D'abord, l'algèbre CCS (*Calculus of Communicating Systems*) [Milner, 1980] permet de prendre en compte le non-déterminisme des processus. Un système communicant y est vu comme un ensemble d'automates non-déterministes qui interagissent par synchronisation. Ensuite, le π -calcul ajoute la notion de « mobilité » [Milner and Parrow, 1992]. La problématique étudiée est ainsi la communication de plusieurs processus au travers d'un réseau dont la topologie change régulièrement. Enfin, les *Mobile Ambients* [Cardelli, 2000] introduisent la notion de localisation. Les processus sont localisés les uns par rapport aux autres et ont la capacité de se mouvoir.

L'idée est alors d'adapter ces algèbres à l'étude de systèmes biologiques. Ainsi, les molécules, les protéines, les cellules... peuvent être vues comme des processus. Les interactions entre les différentes entités sont alors vues comme les communications entre les processus. Il devient ainsi possible de raisonner de manière abstraite, sur des réseaux d'interactions concurrents et de produire des preuves formelles sur les modèles de processus biologiques, là où l'intuition humaine atteint ses limites.

Depuis 2001, différents chercheurs exploitent les algèbres de processus pour formaliser des processus biologiques. Certains utilisent les algèbres déjà existantes comme le π -calcul [Regev et al., 2001] ou sa version stochastique (pour une simulation au niveau mésoscopique, voir 2.3.5.3) [Regev et al., 2001; Regev and Shapiro, 2004]. Le π -calcul est de préférence choisi pour formaliser les réactions moléculaires (figure 2.5). D'autres construisent de nouveaux formalismes inspirés des anciens. Ainsi, le κ -calcul [Danos and Laneve, 2004], aisément transcodable en π -calcul, est un peu plus élégant que ce dernier pour décrire les interactions entre protéines. De la même manière, le langage Brane-Calculi [Cardelli, 2004] permet de décrire facilement les divers comportements des membranes des cellules et de tous ses constituants.

Un des intérêts de ces algèbres, c'est de pouvoir faire de la simulation. Actuellement, l'introduction de calculs stochastiques semble encore être une limite pour mener à bien les méthodes de preuves formelles qui ne les prennent pas encore en compte.

2.2.3 Les réseaux de Petri

Les réseaux de Petri (RdP), autres grands classiques des outils de l'informaticien, sont également mis à contribution pour modéliser les réseaux moléculaires. Inventés par Carl Adam Petri, en 1962 [Petri, 1962], ils sont un formalisme graphique et mathématique qui permet de

```

System ::= Enzyme | Substrate .
Enzyme ::= (new rel_s, rel_p) .
           bind_s ! {rel_s,rel_p} , EX(rel_s,rel_p) ;
           bind_p ! {rel_s,rel_p} , EX(rel_s,rel_p) .
EX(release_s,release_p) ::= release_s ! [ ] , Enzyme ;
                           release_p ! [ ] , Enzyme .
Substrate ::= bind_s ? {erel_s , erel_p} , X(erel_s , erel_p) .
Product ::= bind_p ? {erel_s , erel_p} , X(erel_s , erel_p) .
X(rel_es,rel_ep) ::= rel_es ? [ ] , Substrate ;
                  rel_ep ? [ ] , Product .

```

FIG. 2.5: Un exemple de réaction enzymatique (Michaelis-Menten) décrit en π -calcul, issu de [Regev and Shapiro, 2004]. « | » signifie « et », « ; » signifie « ou », « , » signifie « puis », « ? » signifie « émettre » et « ! » signifie « recevoir ». La réaction est du type : $E + S \rightleftharpoons EP \rightarrow E + P$. Initialement, on considère un système à deux processus : **Enzyme** et **Substrat**. Le processus **Enzyme** donne naissance aux processus fils **EX** et **X** qui représente les deux parties du complexe ES . Le processus **X** crée alors un processus **Produit** ou **Substrat**.

représenter des systèmes (informatiques, industriels) travaillant sur des variables discrètes. Un RdP classique se compose de *places*, de *transitions*, d'*arcs* et de *jetons*. L'idée de Reddy *et al.* [Reddy et al., 1993] est de faire correspondre une place à une population de molécules d'une espèce chimique, une transition à une réaction, un arc à un coefficient stochiométrique, et finalement, un jeton à une molécule. Ainsi, il devient possible de décrire qualitativement un réseau moléculaire au niveau microscopique (voir section 2.3.5.1).

L'état d'un système représenté par RdP correspond à la position des jetons dans les places du réseau. Suivant cet état, certaines transitions peuvent être valides. Le franchissement d'une transition valide est un événement indivisible qui entraîne le déplacement, la création ou la suppression de jetons. Un franchissement d'étape correspond donc à un changement d'état. Lorsque plusieurs transitions sont simultanément valides, il est impossible de déterminer laquelle sera franchie. Le formalisme des RdP n'est pas déterministe, il y a plusieurs états futurs possibles. Si on étend le formalisme aux RdP stochastiques, il devient alors possible de prendre en compte les différentes probabilités de réaction entre les molécules et de décrire quantitativement un réseau moléculaire biochimique au niveau mésoscopique (voir section 2.3.5.3).

Plus récemment est apparue l'utilisation en biologie des RdP hybrides [Doi et al., 2004]. Les réseaux hybrides permettent d'intégrer des RdP continus et fonctionnels. Les places discrètes sont ainsi transformées en places continues qui représentent les concentrations d'espèces moléculaires. Les transitions continues peuvent être associées à des fonctions et possèdent une fréquence d'activation. Un RdP continu fonctionnel permet en fait de décrire un système de résolution numérique d'un système d'équations différentielles. Ce système d'équations pouvant décrire quantitativement un réseau moléculaire biochimique (figure 2.6) au niveau macroscopique (voir section 2.3.5.2). L'intérêt des systèmes hybrides est donc de mélanger les réseaux discrets et continus.

Finalement, il ressort que les réseaux de Petri hybrides et fonctionnels permettent la formalisation qualitative et quantitative de réseaux d'interactions moléculaires en systèmes dynamiques comportant plusieurs niveaux de modélisation. Le formalisme permet, en principe, l'étude qualitative de la structure et de la topologie. Grâce à un logiciel utilisant ce formalisme, GON [Nagasaki et al., 2003], il est possible d'observer l'évolution quantitative du comportement dynamique. Plusieurs études montrent qu'il est possible de retrouver par la

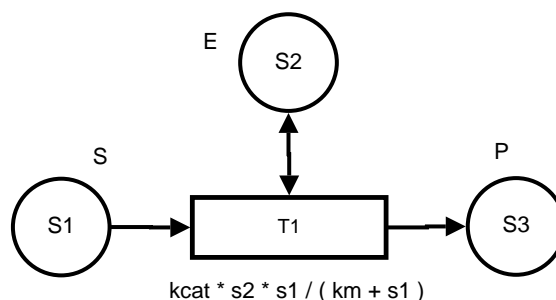


FIG. 2.6: La figure (d'après [Chen et al., 2002]) représente un exemple de réaction enzymatique (transition T1). Cette fois, la réaction est de la forme : $E + S \rightarrow E + P$. L'enzyme (S2) transforme ici, un substrat (S1) en un produit (S3). Le formalisme utilisé est un réseau de Petri continu.

simulation des résultats connus. Il est également intéressant de constater que l'utilisation de réseaux hybrides, impliquant l'utilisation de réseaux continus, limite en grande partie l'application de l'arsenal d'outils de preuves qui sont associés aux réseaux de Petri (vivacité, monotonie, caractère borné, blocage...).

2.2.4 Les logiques temporelles

Nous présentons finalement, une troisième méthode de formalisation des réseaux d'interactions moléculaires que nous avons choisie de décrire dans cette section. Elle se base sur l'utilisation de logiques temporelles à travers l'application BIOCHAM, de l'INRIA. Les logiques temporelles sont des extensions de la logique conventionnelle (propositionnelle) qui intègrent de nouveaux opérateurs exprimant la notion du temps (logique modale). Les opérateurs temporels permettent ainsi de formuler des propriétés sur les exécutions. Selon les logiques, les exécutions sont soit des séquences d'états, soit des arbres, qui représentent l'évolution de systèmes. La différence entre les logiques temporelles provient de l'ensemble des opérateurs temporels qui peuvent être utilisés et des objets sur lesquels ils sont interprétés (séquences ou arbres d'états).

Ces logiques sont donc utilisées pour énoncer formellement des propriétés sur les exécutions du système étudié. La même idée qui a conduit à utiliser les réseaux de Petri ou les algèbres de processus conduit naturellement à utiliser ces logiques pour raisonner sur la dynamique des réseaux.

L'application BIOCHAM [Fages et al., 2004] possède d'une part un langage de description de processus biologiques présenté comme une version simplifiée du π -calcul permettant la description multi-échelles des processus. D'autre part, il utilise un langage de description de propriétés biologiques, élaboré à partir de différentes logiques temporelles. Les propriétés doivent correspondre à des résultats d'expérimentations biologiques que l'on utilise pour construire un modèle. Suivant l'échelle de description des processus à laquelle on se situe, trois logiques sont utilisées : CTL, LTL, PLTL.

La logique CTL (*Computation Tree Logic*) permet de raisonner sur un arbre infini d'états et de transitions. Elle utilise des opérateurs pour les branchements (choix non-déterministes) et pour le temps (les transitions). Les états sont définis par un ensemble de variables binaires ; ce qui convient bien à la description de propriétés sur les réseaux binaires d'interactions.

BIOCHAM fournit un langage de description de règles basées sur la logique CTL pour

décrire certaines propriétés attendues du réseau étudié (états atteignables, états stables, oscillations...). Il fournit également la possibilité de définir un certain nombre d'états initiaux possibles. Ensuite, un mécanisme de *model checking* permet de valider (ou non) l'ensemble des règles établies. Finalement, un algorithme d'apprentissage peut affiner le modèle du réseau pour le faire correspondre aux propriétés décrites.

La logique LTL (*Linear Temporal Logic*) est utilisée pour donner des propriétés sur des modèles décrits au niveau macroscopique (voir section 2.3.5.2) ; c'est-à-dire, lorsqu'on étudie les réseaux pour lesquels on considère que les états sont des concentrations d'espèces moléculaires. Dans ce cas, le langage de description de processus définit un système d'équations différentielles déterministes. Les opérateurs de branchement n'ont alors plus lieu d'être. La logique LTL ne les utilise donc pas et ne considère alors plus l'aspect arborescent, mais séquentiel du système. Des règles concernant les propriétés du système peuvent être décrites de la même manière qu'avec la logique CTL. La correction de ces règles peut également être vérifiée avec un algorithme de *model checking* en les appliquant sur la solution numérique du système. BIOCHAM implémente une méthode de Runge-kutta et une méthode de Rosenbrock (voir section 2.1.1) qui permettent à partir de l'état initial de générer une solution. Un algorithme d'apprentissage associé à l'algorithme de *model checking* permet de déterminer automatiquement un paramètre non défini (coefficient cinétique, constante d'équilibre, coefficient d'inhibition...). Cet outil intéressant permet donc d'automatiser une tâche fastidieuse que tout modélisateur doit couramment réaliser.

La logique PLTL (*Past Temporal Linear Logic*) est un fragment de la logique PCTL (*Probabilistic Computational Tree Logic*). PCTL est obtenue en modifiant certains opérateurs de CTL pour s'intéresser aux réseaux ayant une sémantique stochastique. L'application BIOCHAM utilise un algorithme de type Gillespie (voir 2.3.5.3) pour résoudre les systèmes de processus en considérant des populations de molécules. L'équipe de BIOCHAM travaille actuellement sur l'utilisation de la logique PLTL pour définir des règles pour le niveau de modélisation mésoscopique (voir section 2.3.5.3).

2.2.5 Conclusions

Les trois méthodes présentées montrent comment l'« informatique pour la biologie » s'est engagée dans un travail d'élucidation des processus biologiques en termes biochimiques à l'échelle moléculaire.

En premier lieu, nous remarquons que les formalismes utilisés ont l'avantage de permettre la description, sans ambiguïté, de la dynamique des réseaux. Cette description constitue un modèle. Dans les trois méthodes, le modélisateur peut choisir entre plusieurs échelles de description : discrète, stochastique ou continue.

La volonté originelle de ces travaux est de permettre le raisonnement *a priori*, sans simulation du modèle. Malheureusement, en ce qui concerne les réseaux de Petri et les algèbres de processus, nous remarquons que la volonté d'augmenter l'expressivité du formalisme a pour conséquence de limiter les possibilités de preuves. En effet, la prise en compte de phénomènes stochastiques ou continus rend inutilisable une grande partie des outils de raisonnement automatique qui s'accommode mieux des systèmes discrets. Nous pouvons voir cette limite comme un verrou scientifique qui semble être en bonne voie de déblocage grâce à l'application BIOCHAM.

Ensuite, les modèles étant décrits par ces formalismes, il est possible d'appliquer des algorithmes pour simuler leurs comportements. Il s'avère donc que les formalismes qui, au départ,

étaient destinés à faire du raisonnement, peuvent également servir à simuler les systèmes. Cette possibilité est intéressante car elle permet de compléter par la simulation, l'analyse *a priori*. Nous constatons cependant que nombre de travaux (par exemple [Kuttler and Niehren, 2006], [Troncale et al., 2006]) se focalisent exclusivement sur l'aspect simulation ; ce qui est contradictoire avec les objectifs initiaux de ces formalismes justifiés et construits avec la contrainte du raisonnement *a priori*. Nous pouvons supposer que d'autres formalismes libérés de cette contrainte seraient plus expressifs et surtout plus ergonomiques.

Enfin, nous constatons qu'il est souvent possible de passer d'un formalisme à l'autre. Ainsi, un réseau de Petri peut s'exprimer par l'algèbre de processus. BIOCHAM utilise une version simplifiée de l'algèbre de processus. Il est donc possible d'entrevoir des interfaces entre les quelques formalismes qui échapperont à la sélection naturelle de la communauté.

Les trois méthodes se focalisent uniquement sur les réseaux d'interactions chimiques. Par rapport aux réflexions sur le changement de paradigme en biologie (section 1.5) et à la volonté de créer « le » formalisme du vivant, elles apparaissent plus comme des formalismes de la chimie et n'apportent pas vraiment de spécificité biologique. De plus, la variété des modèles d'interactions chimiques complique l'utilisation d'outils de preuves *a priori*. Cette difficulté pourrait devenir vraiment insurmontable si on décidait d'introduire des interactions de natures différentes dans les réseaux (mécaniques, physiques...). Intéressons-nous donc maintenant plus à l'aspect simulation qui se dégage de l'utilisation de l'informatique pour la biologie.

2.3 L'avènement de la simulation numérique

Ce chapitre traite de la prise en compte de la complexité du vivant dans les applications informatiques. Il est temps de faire une pause à mi-parcours de cet état de l'art pour en introduire plus clairement la suite. À la lecture du début de cette thèse, nous avons pu constater que la biologie utilise différents moyens d'études pour faire évoluer l'état des connaissances sur le vivant. Cette section résume ces méthodes et nous focalise sur l'une d'elle : la simulation, qui nous intéresse plus particulièrement.

2.3.1 L'expérimentation *in vivo*

La biologie est en premier lieu, une science expérimentale (voir section 1.1). L'expérimentation *in vivo* est la méthode de base en biologie. Elle est nécessaire à toute modélisation, toute hypothèse sur le vivant. Aujourd'hui encore, la validation d'un nouveau traitement médical doit obligatoirement passer par des tests *in vivo*.

L'objet de ce type d'expérience, c'est l'organisme. En effet un organisme est vu comme une instance particulière de l'espèce à laquelle il appartient. Il modélise ainsi cette espèce de la manière la plus fiable que l'on puisse considérer. Ainsi une batterie de tests sur un ensemble de patients suffisamment représentatif permet de valider une hypothèse sur une pathologie. Parfois une espèce peut être également considérée comme un modèle approché d'une autre espèce. Les expériences *in vivo* sont par exemple pratiquées sur les souris pour faire avancer les connaissances sur des pathologies humaines. Nous reviendrons sur la notion d'expérience dans le chapitre suivant (voir section 3.1.2).

2.3.2 L'expérimentation *in vitro*

Les expériences *in vivo* peuvent dans certaines situations poser des problèmes techniques ou éthiques. Le biologiste doit alors trouver des méthodes alternatives pour contourner les contraintes qui s'imposent à lui. Dans ce cas, le modèle devient une représentation artificielle du phénomène ou de l'objet étudié. Les méthodes alternatives consistent à tester le comportement de cette représentation sous l'effet d'actions que l'on peut exercer sur elle. C'est le cas des expériences *in vitro* dans lesquelles on étudie un échantillon ou une maquette physique construite par analogie avec le système réel. L'apport technique des physiciens et des chimistes à l'époque de l'émergence de la biologie moléculaire a été considérable dans ce domaine (voir sections 1.2 et 1.3). Cependant, l'explosion des données provoquée par les avancées de l'expérimentation conduit désormais, à mettre l'accent sur d'autres méthodes plus théoriques pour interpréter les résultats des expériences.

2.3.3 Le raisonnement *in abstracto*

Les expériences *in vivo* et *in vitro* permettent de construire des modèles théoriques des systèmes ou phénomènes étudiés, en interprétant les données obtenues. Il faut ensuite étudier ces modèles théoriques pour formuler de nouvelles hypothèses, ou pour préciser et produire de nouvelles connaissances.

De plus, l'observation d'un système réel, d'un échantillon ou d'une maquette physique peut, dans certains cas, interférer avec le phénomène étudié. En outre, même *in vitro*, les moyens technologiques sont souvent trop limités pour permettre l'observation détaillée du phénomène. Dans ce cas, on a recours à la théorie qui permet de raisonner sur des modèles abstraits.

Nous pouvons appeler raisonnement *in abstracto*, le raisonnement sur un modèle théorique formel. C'est ce que nous avons présenté dans les sections précédentes (voir sections 2.1 et 2.2). Les méthodes de raisonnement mathématiques ou logiques permettent de démontrer certaines propriétés de modèles théoriques. Chaque méthode vise plus ou moins à définir une théorie permettant le raisonnement dans le monde des idées, répondant ainsi aux questions posées par le changement de paradigme en biologie (section 1.5).

2.3.4 Les calculs *in silico*

Nous avons pu remarquer dans les sections 2.1 et 2.2 que les théories permettant le raisonnement *in abstracto* font souvent appel à l'automatisation des processus de démonstration. Cette automatisation est due à la complication (et non à la complexité) des tâches à réaliser. De plus, nous avons observé que les méthodes de raisonnement sont associées à des outils de simulations numériques.

Notons également que lorsque l'analyse mathématique du modèle théorique d'un système est impossible, ou tout au moins trop difficile, le scientifique est amené à le simuler. Évidemment, nous parlons ici, de la simulation numérique. La simulation permet, de mettre en œuvre le système à partir de conditions initiales, pour en étudier la dynamique.

Il est d'usage de regrouper la simulation, l'automatisation des processus de démonstration et plus généralement, l'exécution de tout algorithme utile à l'étude du vivant sous l'appellation « calculs *in silico* ».

Nous avons présenté dans les sections précédentes les formalismes qui permettent l'analyse

mathématique (sans simulation). Mais, cette présentation nous a montré que la dynamique des populations, la théorie du chaos, les fractales, l'algèbre de processus, les réseaux de Petri et l'application BIOCHAM, faisaient intervenir la simulation numérique conjointement à cette analyse. Nous pouvons donc distinguer deux types d'analyse :

- l'analyse « logique » du modèle pour découvrir des propriétés, *a priori*, sans le simuler ;
- l'analyse « numérique » de la dynamique d'un système à partir des résultats de sa(ses) simulation(s) numérique(s).

Finalement, nous affirmons que la simulation demeure la composante principale des calculs *in silico*. En effet, dans la plupart des études biologiques utilisant l'outil informatique, la simulation représente la seule partie automatisée du raisonnement *in abstracto*².

Forts de cette constatation, nous allons nous intéresser plus particulièrement à la partie des recherches *in silico* qui n'aspire pas à la pratique de l'analyse logique du modèle *a priori*, mais fournit des moyens de simulation aux biologistes. Le soin de l'analyse numérique est ainsi laissé à l'expertise du biologiste. Ainsi par la suite, nous utiliserons souvent les termes *in silico* et « simulation » comme synonymes.

2.3.5 La granularité des modèles

Au long de la section 2.2 sur les formalismes « bio-logiques », nous avons déjà remarqué que les simulations font appel à des modèles décrits à différents niveaux. Trois termes qui servent à l'origine à désigner une échelle d'observation de phénomènes physiques, peuvent être utilisés pour décrire la granularité d'un modèle, suivant le point de vue du modélisateur. Il s'agit des échelles *microscopique*, *mésoscopique* et *macroscopique*.

2.3.5.1 Microscopique

Le niveau microscopique consiste à modéliser individuellement chaque entité qui compose un système. Il fait appel à une description locale des modèles. Le comportement global du système pourra alors émerger de la simulation de l'ensemble des modèles microscopiques. L'échelle microscopique est le moyen le plus précis pour décrire un phénomène. Bien souvent les méthodes utilisées ne sont pas déterministes du fait de leur caractère aléatoire. En général, les modèles sont spatialisés. Les systèmes multi-agents se prêtent bien à ce type de modélisation (section 2.4.5).

Si on s'intéresse à la cinétique chimique, les dimensions temporelles et spatiales sont à l'échelle de la molécule (nanosecondes, angströms). À cette échelle, chaque espèce moléculaire est représentée individuellement. La simulation consiste alors à faire se mouvoir ces différentes espèces moléculaires au sein d'un volume et à les faire éventuellement réagir lorsqu'elles entrent en collision. L'espace doit obligatoirement être pris en compte, c'est pourquoi on ne distingue pas modélisation temporelle et modélisation spatio-temporelle. L'espace des états est donné par la liste des positions et éventuellement orientations des différentes espèces moléculaires. Les trajectoires suivies peuvent être discrètes ou continues. En effet, les déplacements des espèces moléculaires sont reproduits par des marches aléatoires qui se font, soit dans un espace discret, soit dans un espace continu. Dans les deux cas, les déplacements ainsi que les réactions induites par les collisions entre les diverses espèces moléculaires sont donnés par des lois probabilistes : le comportement du système est alors stochastique. Les

²Pour être plus précis, nous devons distinguer l'analyse numérique, du raisonnement *in abstracto*. Le calcul numérique entraîne des erreurs telles les approximations sur les réels. Il n'est donc plus strictement équivalent au monde des idées. Le calcul *in silico* repose sur le raisonnement *in abstracto*.

méthodes utilisées sont donc de type « Monte Carlo » [Takahashi and Tomita, 2005]. Compte-tenu de la puissance de calcul requise par ce type de simulations, les techniques et méthodes informatiques pour la simulation de la cinétique biochimique à l'échelle microscopique sont pour l'instant peu développées, le travail de recherche en informatique dans ce domaine en est encore à son début.

2.3.5.2 Macroscopique

La simulation d'un système modélisé au niveau microscopique est souvent démesurément coûteuse en puissance de calcul, du fait du nombre d'entités considérées. Pour pallier ce problème les modélisateurs ont recours à la simulation à l'échelle macroscopique. Ce niveau de granularité, considère le comportement global d'une population.

Dans l'exemple de la réaction chimique, cela revient à modéliser l'évolution globale et continue des populations de molécules. En effet, il est souvent possible d'extrapoler le comportement global d'une population en faisant la moyenne des comportements observés au niveau microscopique. De plus, en chimie, ce sont bien les concentrations que l'on observe expérimentalement et non le comportement individuel des molécules. La plupart des lois exprimées et utilisées en sciences physiques sont ainsi de nature macroscopique, car c'est l'échelle naturelle d'observation de la plupart des phénomènes. L'échelle macroscopique est le domaine quasi exclusif des équations différentielles. Les méthodes de résolution numérique présentées dans la section 2.1.1 et la dynamique des populations (voir section 2.1) s'inscrivent dans ce cadre.

2.3.5.3 Mésoscopique

Parfois, il est impossible de décrire correctement un phénomène au niveau macroscopique car la quantité d'entités mises en jeu au niveau microscopique est trop faible pour extrapoler une loi globale d'évolution de la population sans s'intéresser aux interactions locales. L'hypothèse de continuité n'est plus valable. Cette situation conduit naturellement à affiner la granularité du modèle pour l'étudier au niveau microscopique. Mais, la quantité d'entités peut tout de même s'avérer assez élevée pour interdire une simulation à l'échelle microscopique demeurant trop coûteuse. De plus, il est souvent très difficile de modéliser l'espace dans lequel se déroulent les phénomènes locaux.

L'échelle mésoscopique permet alors de proposer une solution pour simuler de tels phénomènes en se plaçant à un niveau intermédiaire entre le modèle d'individu et celui de population.

Dans l'exemple de la modélisation de la cinétique chimique, l'utilisation des équations différentielles repose sur un certain nombre d'hypothèses. On suppose par exemple que les concentrations d'espèces chimiques évoluent de manière continue et déterministe. Il se trouve que la faible quantité de molécules impliquées dans les réactions intra-cellulaire (quelques centaines, ou même une seule si on parle d'un brin d'ADN) compromet ces hypothèses [Arkin et al., 1998].

La simulation « stochastique exacte » est une méthode probabiliste pour réaliser la simulation à ce niveau de modélisation. L'algorithme de Gillespie [Gillespie, 1976] tient toujours compte du nombre de molécules mais ne se préoccupe ni de leur position ni de leur moment cinétique. Les variables représentant les nombres de molécules varient discrètement. On calcule la probabilité de collision entre deux molécules par unité de temps et de volume. On parle alors de probabilité de réaction. A ce niveau, la modélisation et la simulation de larges systèmes sont plus aisées qu'au niveau microscopique. Divers algorithmes ont été pro-

posés successivement pour la simulation stochastique exacte en milieu homogène [Gibson and Bruck, 2000; Cao et al., 2004] et en milieu hétérogène [Ander et al., 2004; Elf and Ehrenberg, 2004].

2.3.6 Vers une simulation multi-échelles

Il faut garder à l'esprit que la notion de granularité n'a de sens que par rapport à l'objet modélisé. Ainsi, si nous modélisons un système de cellules et que chaque cellule est vue comme un réacteur chimique indépendant mettant en œuvre un système d'équations différentielles, nous observons à la fois une granularité microscopique du système cellulaire et une granularité macroscopique des phénomènes moléculaires. Pour aller plus loin, la rigueur impose de modéliser certaines réactions au niveau mésoscopique car les concentrations au sein de la cellule ne sont pas suffisantes. Ce faisant, le modèle mélange les visions macroscopique et mésoscopique du niveau moléculaire. Ce simple exemple nous montre qu'une simulation de système vivant est amenée à considérer et à intégrer des phénomènes qui se réalisent à plusieurs niveaux de description.

Lorsqu'un modèle fait intervenir plusieurs granularités dans sa description, la simulation doit mettre en œuvre des calculs qui se passent à des échelles de temps et d'espace très différentes. Par exemple, une résolution au niveau macroscopique d'une réaction enzymatique se calcule avec un pas de temps compris entre 10^{-3} et 10^{-6} seconde; alors qu'une simulation de type microscopique de cette modélisation fait intervenir des pas de temps de 10^{-9} à 10^{-12} seconde. La cohabitation d'échelles de calculs très différentes au sein des simulations conduit à la notion de « simulation de modèle multi échelles ».

Nous avons choisi de nous concentrer sur les outils permettant la simulation. La simulation, calculs *in silico*, vient compléter le raisonnement *in abstracto* et les expérimentations *in vivo* et *in vitro*. La simulation multi-échelles est une propriété essentielle permettant de mélanger les niveaux de description microscopique, mésoscopique et macroscopique. Si nous poussons plus avant cette idée de simulation multi-échelles, nous pouvons commencer à entrevoir la notion de « simulation multi-modèles ». La section suivante est dédiée aux outils la mettant en œuvre.

2.4 Types d'architectures pour les simulations *in silico*

« A scientist ... is usually expected not to write on any topic of which he is not a master... We are only now beginning to acquire reliable material for welding together the sum total of all that is known into a whole [but] it has become next to impossible for a single mind fully to command more than a small specialised portion... I can see no other escape from this dilemma than that some of us should venture to embark on a synthesis of facts and theories, albeit with second-hand and incomplete knowledge ... and at the risk of making fools of ourselves » (Erwin Schrödinger [Schrödinger, 1944])

Il nous paraît maintenant évident que la modélisation et la simulation de mécanismes du vivant doivent être multi-échelles. En effet, les formalismes pour le raisonnement sur les réseaux biochimiques sont déjà construits sur ce principe (voir section 2.2) en faisant cohabiter différentes granularités dans les modèles. Il serait dommage de posséder un paradigme de

simulation moins expressif que ces formalismes fortement contraints par leur volonté de faire des preuves de propriétés.

De plus, la modélisation multi-échelles permet de considérer d'une part, une vision biologique et systémique d'un système dans son environnement mettant en œuvre des connaissances grossières ou approximatives, et d'autre part, des connaissances très précises sur certains points comme par exemple sur la cinétique des réactions chimiques impliquées dans ce système [Sorger, 2005]. Une des raisons qui rendent incontournable cette association de points de vue, est qu'il est impossible dans la très grande majorité des cas, de décrire parfaitement la chimie d'un système ou d'un phénomène impliqué dans le vivant. La biologie réputée comme une science riche de données est paradoxalement handicapée par le manque d'informations précises et quantitatives concernant la chimie et la cinétique chimique des phénomènes étudiés. Et, quand bien même aurions-nous accès aux informations permettant de décrire de manière exhaustive la chimie d'un système telle la cellule, nous serions incapables de le simuler, tant le modèle serait gigantesque. Cette tâche rejoindrait la vision mécaniste obsolète qui rêve d'écrire l'équation complète et déterministe de la dynamique d'un monde objectif.

2.4.1 La simulation « multi-modèles »

Dans le même ordre d'idée, il serait réducteur de vouloir ne modéliser que la chimie en milieu homogène. La seule dimension temporelle et la seule nature chimique des phénomènes sont insuffisantes pour s'intéresser aux systèmes complexes biologiques en général. Les discussions récentes sur l'étude du vivant par la modélisation informatique montrent que l'espace est une dimension fondamentale [Takahashi and Tomita, 2005]. Certains facteurs chimiotactiques, la forme des organes, la forme des cellules, l'environnement... jouent également un rôle dans l'organisation des processus.

Enfin, la simulation multi-modèles permet de faire cohabiter les approches descendantes et ascendantes de la modélisation. L'élaboration *top-down* ou descendante d'un modèle correspond à l'image du sculpteur taillant une pierre. Le modèle est d'abord défini grossièrement dans sa globalité. Chaque étape vise ensuite à réduire d'un cran la granularité ou l'imprécision du modèle pour évoluer vers un modèle de plus en plus précis. La construction ascendante ou *bottom-up* de modèles ressemble plus à un jeu de Legos. Le modélisateur utilise alors les briques de base pour construire des composants qui eux mêmes serviront à constituer un modèle plus global... La simulation multi-modèles permet d'appliquer l'une ou l'autre des deux méthodes, de les mélanger en associant des modèles issus de l'une ou de l'autre et finalement de ne plus se poser la question de la méthode ascendante ou descendante.

Il nous reste maintenant à découvrir comment cette idée de simulation multi-modèles peut être mise en œuvre. D'abord, nous considérerons la méthode de la « multi-simulations » permettant la collaboration entre les simulateurs et les différents acteurs de la modélisation, grâce à l'usage d'une norme, SBML. Ensuite, nous aborderons un exemple montrant comment un simulateur, VirtualCell, peut aborder le multi-modèles, par l'usage des méthodes classiques. Enfin nous nous orienterons vers les architectures objets et agents qui nous semblent plus adéquates.

2.4.2 SBML

2.4.2.1 La « multi-simulations »

Une façon d'appréhender la simulation multi-modèles, est de la considérer en tant que « multi-simulations ». En effet, le domaine de la biologie systémique fournit à la communauté plusieurs simulateurs utilisant différentes méthodes de résolutions numériques. Au cours de cet état de l'art nous en avons déjà référencé un certain nombre. Il en ressort que chaque simulateur possède des propriétés qui le rendent plus ou moins efficace pour tel ou tel type de problème. Le chercheur dont le sujet d'étude est multi-modèles, peut alors être amené à choisir l'outil le plus adapté pour traiter certaines sous-unités de son modèle. De plus, il peut être intéressant, pour valider un résultat numérique, d'effectuer la simulation d'un même modèle avec plusieurs méthodes de résolution différentes.

Sachant que pour son fonctionnement, un simulateur doit nécessairement manipuler une description des données constituant le modèle numérique, il faut pouvoir réutiliser cette description avec un autre logiciel avec un minimum de complications. La constitution d'un modèle complet de phénomène complexe nécessite la réutilisation, la comparaison, la modification ou l'association [Snoep et al., 2006] de modèles déjà établis par d'autres chercheurs. D'où le besoin quasi évident de posséder une norme commune d'échange et de description de données biologiques partagées par les différents simulateurs. Notons également qu'il serait dommage de perdre les modèles implémentés avec un simulateur lorsque ce simulateur n'est plus maintenu, donc plus utilisable.

La tendance a été ces dernières années, à l'utilisation de langages du type XML (*eXtended Markup Language*) pour décrire les données. La communauté de la biologie systémique n'y a pas dérogé. Ainsi, différents langages ont vu le jour : CellML [Lloyd et al., 2004], SBML, BioPAX, PSI MI [Stromback and Lambrix, 2005]... L'actualité semble montrer que SBML a réussi à s'imposer comme la norme internationale permettant de décrire les données relatives aux systèmes de réactions biochimiques.

2.4.2.2 La norme

L'objectif de la norme est de permettre la séparation du modèle numérique et de la méthode de résolution que l'on utilisera pour le simuler. Ainsi, SBML [Hucka et al., 2003; Finney and Hucka, 2003; Hucka et al., 2004] (*System Biology Markup Language*) est un langage à balises, de type XML, qui par la création de balises adéquates, est adapté aux besoins des biologistes, pour le développement de modèles de systèmes biologiques. Il permet de représenter des réseaux de réactions biochimiques, comme par exemple des cascades métaboliques ou des voies de signalisations cellulaires, de façon assez simple.

Un fichier SBML (*level 2*) comporte quelques balises fondamentales :

<model> est la balise racine du modèle décrit dans le fichier.

<compartment> permet de fragmenter l'espace dans lequel le système biologique prend place.

Un compartiment possède 2 ou 3 dimensions, et surtout une taille.

<species> est la représentation d'une espèce chimique présente dans un compartiment et pouvant prendre part aux réactions.

<reaction> est la modélisation des réactions chimiques, qui vont modifier les caractéristiques des espèces chimiques selon des règles définies ici.

<rule> est la modélisation de phénomènes qui vont modifier les caractéristiques des espèces chimiques selon des règles définies ici.

Dans une balise `reaction`, se trouve une règle mathématique, ainsi que des références vers les espèces chimiques, sous différents noms. Les Species peuvent en effet être soit `reactant`, soit `product`, soit `modifier` au sein d'une réaction. La différence entre ces balises se fait au niveau des effets de la réaction sur les espèces : un `reactant` verra sa concentration diminuer au cours de la réaction, au contraire d'un `product` qui la verra augmenter, alors que celle d'un `modifier` restera inchangée (c'est typiquement le cas d'une enzyme).

Le site internet de référence de SBML est <http://www.sbml.org>. La page d'accueil y référence une centaine de logiciels utilisant cette norme. Les différentes personnes impliquées dans la communauté SBML utilisent des méthodes d'analyse et de simulation variées (équations différentielles, π -calcul, algorithme de Gillespie...). La figure 2.7 montre un exemple de réaction enzymatique décrite en SBML.

2.4.2.3 Éditeurs graphiques

La norme SBML est conçue comme un format d'échange entre logiciels. De ce fait, un modèle SBML n'est pas forcément des plus aisés à rédiger manuellement, d'autant plus que le modélisateur est censé être plus biologiste qu'informaticien. Ainsi, il existe des logiciels offrant des interfaces permettant de définir des modèles SBML. Les plus intéressants sont ceux qui utilisent un langage graphique. Parmi ces logiciels nous pouvons citer deux exemples : `bio sketch pad` [Alur et al., 2002] (figure 2.8) et `CellDesigner` [Kitano et al., 2005] (figure 2.9).

Ces logiciels permettent en plus de la création de modèles, la consultation d'autres modèles importés depuis internet. Ils font partie intégrante de ce que nous appelons la multi-simulations.

Finalement, SBML permet de communiquer les aspects les plus essentiels d'un modèle, d'un chercheur à l'autre et d'un logiciel à l'autre. Pour conclure sur son intérêt, nous remarquons deux problèmes soulevés : souvent, les articles qui publient des modèles entraînent quelques ambiguïtés. D'abord parce que chacun utilise sa propre représentation graphique. Sur ce point les concepteurs de logiciels tentent de proposer des normes de représentation [Kitano, 2003]. Ensuite parce la façon de décrire les modèles dépend souvent de la méthode de simulation. Si un article pointe sur l'adresse d'un modèle SBML disponible en ligne, les ambiguïtés sont levées.

Ce langage initialement conçu pour la « multi-simulations », est également utilisé par des simulateurs multi-modèles classiques.

2.4.3 L'exemple de Virtual Cell

`Virtual Cell` [Slepchenko et al., 2002] est un très bon exemple de plateforme de calcul *in silico* utilisant les méthodes classiques pour l'étude de la cellule. Il s'inscrit dans la pensée suivante : un outil de simulation approprié doit avant tout proposer une interface qui permet au biologiste de modéliser, lancer les calculs et visualiser les résultats dans le but de permettre la comparaison directe avec les résultats expérimentaux (*in vivo* ou *in vitro*). Dans un même temps, l'outil doit être suffisamment sophistiqué pour permettre l'analyse du modèle par les théoriciens. La communication entre théoriciens et expérimentateurs pourrait s'en trouver améliorée.

Le logiciel `Virtual Cell` se place également dans la tendance qui vise à prendre du recul par rapport au niveau d'organisation moléculaire pour s'intéresser au niveau d'organisation

```
<?xml version="1.0" encoding="UTF-8"?>
<sbml level="2" version="A" xmlns="http://www.sbml.org/sbml/level2/version1">
<model name="EnzymaticReaction">

  <listOfCompartments>
    <compartment id="cytosol" size="1e-14"/>
  </listOfCompartments>

  <listOfSpecies>
    <species compartment="cytosol" id="P" initialAmount="0" name="P" />
    <species compartment="cytosol" id="S" initialAmount="1e-20" name="S" />
    <species compartment="cytosol" id="E" initialAmount="5e-21" name="E" />
  </listOfSpecies>

  <listOfReactions>
    <reaction id="RE">
      <listOfReactants>
        <speciesReference species="S"/>
      </listOfReactants>
      <listOfProducts>
        <modifierSpeciesReference species="E"/>
      </listOfProducts>
      <listOfProducts>
        <speciesReference species="P"/>
      </listOfProducts>
      <kineticLaw>
        <math xmlns="http://www.w3.org/1998/Math/MathML">
          <apply>
            <divide>
              <apply>
                <times/>
                <ci>k</ci>
                <ci>E</ci>
                <ci>S</ci>
              </apply>
            <plus/>
            <ci>km</ci>
            <ci>S</ci>
          </apply>
        </math>
      </kineticLaw>
    </reaction>
  </listOfReactions>
</model>
</sbml>
```

FIG. 2.7: Nous reprenons l'exemple de la réaction enzymatique, cette fois décrit avec la norme SBML. La réaction est de la forme : $E + S \rightarrow E + P$.

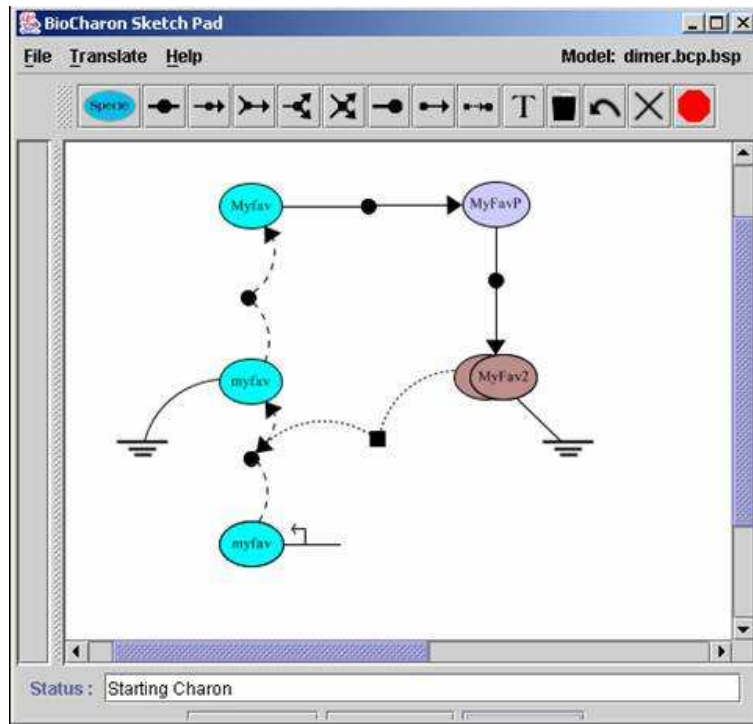


FIG. 2.8: Capture d'écran du logiciel Bio Sketch Pad : http://www.cis.upenn.edu/biocomp/new_html/biosketch.php3.

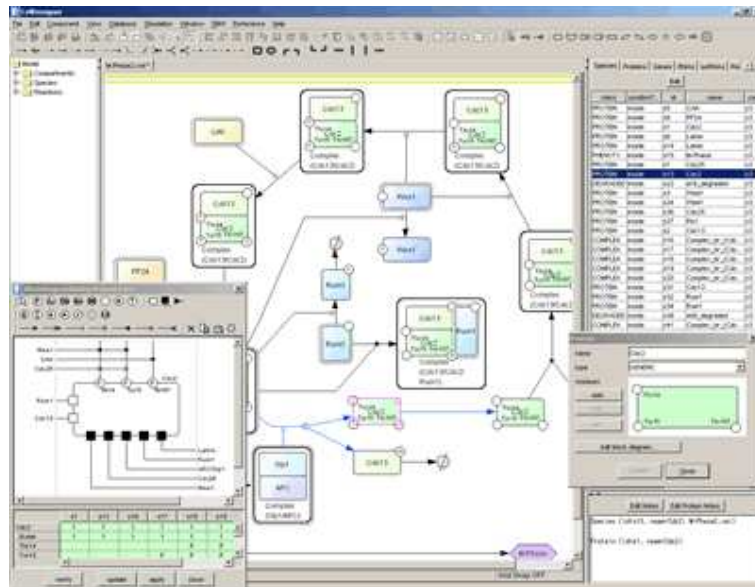


FIG. 2.9: Capture d'écran du logiciel CellDesigner : <http://celldesigner.org/>.

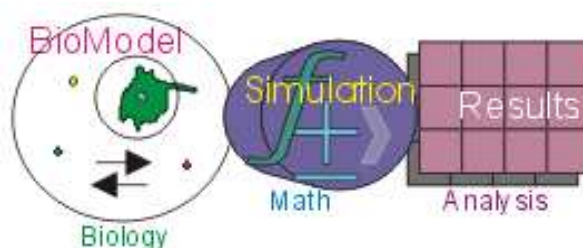


FIG. 2.10: Principe de simulation de Virtual Cell en trois étapes (<http://www.nrcam.uchc.edu/index.html>) : Définition du modèle, création automatique et résolution d'un système mathématique, visualisation des résultats après les calculs.

supérieur qu'est la cellule. Ce changement de niveau est plus intéressant d'un point de vue systémique, puisqu'il permet de glisser des notions purement biochimiques vers des notions plus physiologiques tout en conservant avec rigueur les considérations physico-chimiques.

Dans ce cadre, le modèle de la cellule repose sur des hypothèses biochimiques et électro-physiques (adaptées à l'étude de neurones). L'originalité historique de *Virtual Cell* est la prise en compte de la dimension spatiale dans les modèles. Il est possible d'entrevoir l'impact de la morphologie et de la localisation sur la physiologie de la cellule. La structure en deux ou trois dimensions de la cellule peut être définie à partir de données expérimentales obtenues par microscopie.

Lorsque la diffusion des molécules doit être considérée dans la géométrie complexe de la cellule, des systèmes de « réaction/diffusion » y sont implémentés. *Virtual Cell* est une des seules plateformes de ce type actuellement éprouvée pour la simulation d'équations aux dérivées partielles (EDP). Pour résoudre les systèmes d'EDP, la méthode des éléments finis [Zienkiewicz and Taylor, 1989], celle des différences finies [Langtangen, 2003] ou la méthode des volumes finis [Patankar, 1980] peuvent être mises en œuvre. Chacune de ces méthodes repose sur un maillage de l'espace permettant une résolution locale du système. *Virtual Cell* utilise la méthode des volumes finis [Schaff et al., 2001, 1997] pour résoudre les systèmes de « réaction/diffusion ».

Le logiciel se compose d'une interface client Java, accessible via le site du *National Resource for Cell Analysis and Modeling* (<http://www.nrcam.uchc.edu/>). Elle permet de construire les modèles. Les simulations sont ensuite réalisées sur un serveur distant qui renvoie les résultats de la simulation. Cette démarche permet d'assurer à chaque utilisateur une puissance de calcul suffisante. La figure 2.10 montre le cycle de *Virtual Cell* qui est caractéristique des calculs *in silico*.

La simulation jouée, les résultats spatio-temporels peuvent être visualisés sous la forme de films, d'images, ou de courbes (figure 2.11) et sont alors analysés par le modélisateur.

Il est également possible de déclarer des paramètres comme étant variables et de lancer des batteries de tests pour chaque combinaison de valeurs de ces paramètres, pour en observer l'influence sur les résultats. Cet outil permet d'analyser la sensibilité du modèle aux variations de ces paramètres.

Virtual Cell s'inscrit clairement dans la vision multi-modèles et également dans le cadre de la « multi-simulations » puisqu'il est compatible avec SBML. Il privilégie de plus, l'interface avec le métier de biologiste. Il est regrettable qu'aucun outil de simulation du ni-

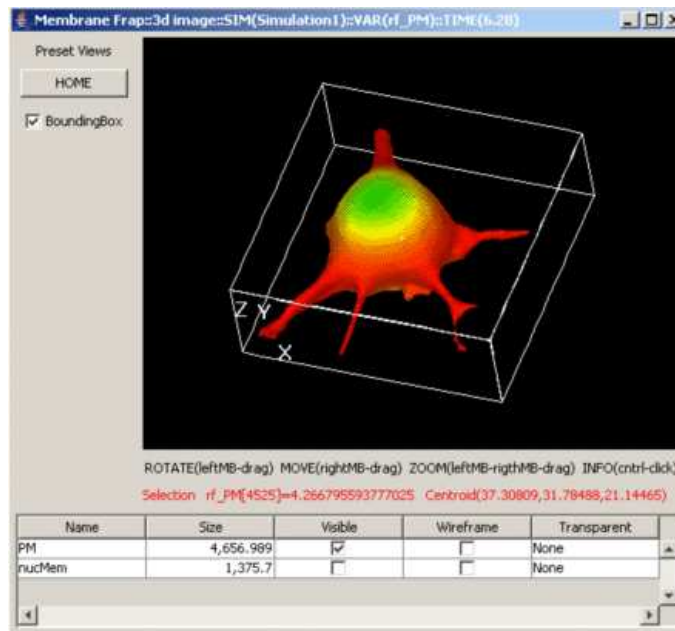


FIG. 2.11: Visualisation des résultats de la simulation avec Virtual Cell.

veau *mésoscopique* n'ait encore été intégré dans la version client du logiciel. Si ce niveau de modélisation est requis il est alors possible de s'adresser à d'autres plateformes de simulation qui le permettent. Il existe d'autres simulateurs du même type : M-CELL, StochSim [Le Novere and Shimizu, 2001], E-CELL [Takahashi et al., 2002], BioSpice [Kumar and Feidler, 2003], The Silicon Cell, A-Cell... Chacun avec leurs spécificités. Tous fonctionnent à peu près suivant le même cycle :

1. Définition d'un modèle biochimique
2. Mathématisation du modèle biologique
3. Résolution numérique du système ainsi généré
4. Visualisation et analyse des résultats obtenus.

Ici, la mise en équations est une étape bien distincte. D'autres constructions informatiques permettent de sortir de ce cycle. C'est le cas de la programmation orientée objet.

2.4.4 Architectures orientées objet

Les paradigmes de programmation de l'informatique, permettent d'entrevoir différentes façons d'appréhender les calculs *in silico*. La solution basique, impérative, est celle que nous venons d'évoquer (voir section 2.4.3) : la mise en équations du système, puis l'exécution d'algorithmes de résolution mathématique. Cette vision relègue en quelque sorte, l'outil de simulation à une simple interface permettant de transcrire un modèle biologique en équations qui seront résolues par un ordinateur « calculatrice ».

Les « nouveaux » outils de l'informatique permettent d'envisager plus finement la construction des modèles numériques et leur simulation. Parmi les exemples que nous avons abordés, les algèbres de processus (voir section 2.2.2) nous ont déjà laissé entrevoir l'intérêt de la programmation fonctionnelle pour l'étude des systèmes biologiques. Ici, nous nous intéressons

plutôt aux paradigmes allant de la programmation orientée objets (POO) aux systèmes multi-agents (SMA). Encore une fois, le but n'est pas d'être exhaustif mais de présenter un panel représentatif du type d'application que l'on peut trouver dans la littérature.

Il est étonnant de constater que ces applications, mais surtout les gens travaillant sur ces questions semblent encore marginaux par rapport à la majorité des outils orientés « résolution mathématique ». Ce constat peut s'expliquer par plusieurs causes :

- La mathématisation vide les problèmes concrets de leur sémantique et permet aux mathématiciens de les traiter dans le cas général, indépendamment de leur domaine scientifique d'origine. Cette pratique est bien adaptée au cloisonnement disciplinaire. Ainsi, nombre de méthodes de résolution mathématique éprouvées sont immédiatement disponibles dans la littérature.
- Les chercheurs du domaine interdisciplinaire de la biologie systémique sont d'origines variées et ont très souvent des parcours atypiques. La maîtrise des paradigmes Objet et Agent requiert en général une formation initiale en informatique (assez récente). Nombre de chercheurs du domaine n'ont pas eu accès à ces paradigmes.
- Il n'y a pas encore eu beaucoup de résultats prouvant directement l'intérêt de ces méthodes en biologie. Les différents travaux réalisés sont souvent taxés de « boxologie ».

Pourtant il nous apparaît que ces concepts sont une voie d'exploration inévitable. L'expérience acquise par des années de conception et de développement de logiciels de plus en plus complexes, peut être réinvestie dans la modélisation et la simulation de systèmes biologiques.

2.4.4.1 UML pour formaliser le vivant

La première idée est d'utiliser les formalismes visuels de conception et d'analyse de logiciels pour décrire le vivant. Dans ce cadre David Harel est un précurseur. Il propose de représenter les modèles biologiques au moyen de diagrammes états/transitions [Harel, 1987]. Les cellules et les organisations multi-cellulaires peuvent ainsi être assimilées à des systèmes réactifs en temps réel [Kam et al., 2001]. Les diagrammes états/transitions font aujourd'hui partie intégrante de la norme UML et peuvent être définis au moyen de logiciels tels que *Rational Rose Real-Time*. Par cette approche, il devient possible de pallier à la faible modularité des modèles à bases d'équations différentielles.

La cellule est alors vue comme un système défini par des combinaisons de ses états internes. Les transitions entre ces états internes sont des conséquences des signaux induits par l'environnement de la cellule. On peut décrire le comportement d'une cellule en répondant aux questions :

- Où se trouve la cellule ?
- Quelles autres entités peut-elle rencontrer ?
- Quelles sortes de signaux peut-elle recevoir ?
- Quel comportement doit-elle produire en sortie ?

Les cellules sont ainsi traitées comme des boîtes noires dans lesquelles n'importe quel comportement peut être défini.

Les travaux autour du projet CellAK (*Cell Assembly Kit*) [Webb and White, 2004] s'inscrivent dans la continuité de ces idées. Le formalisme UML (*Unify Modelling Language*) y est pleinement utilisé. Les classes, l'héritage, la composition... permettent de décrire précisément les entités biologiques (figures 2.12 et 2.13). Les relations entre les différentes classes permettent de se placer à différents niveaux d'organisation. Les aspects dynamiques qui permettent de définir l'évolution dans le temps des modèles sont répartis dans les méthodes des

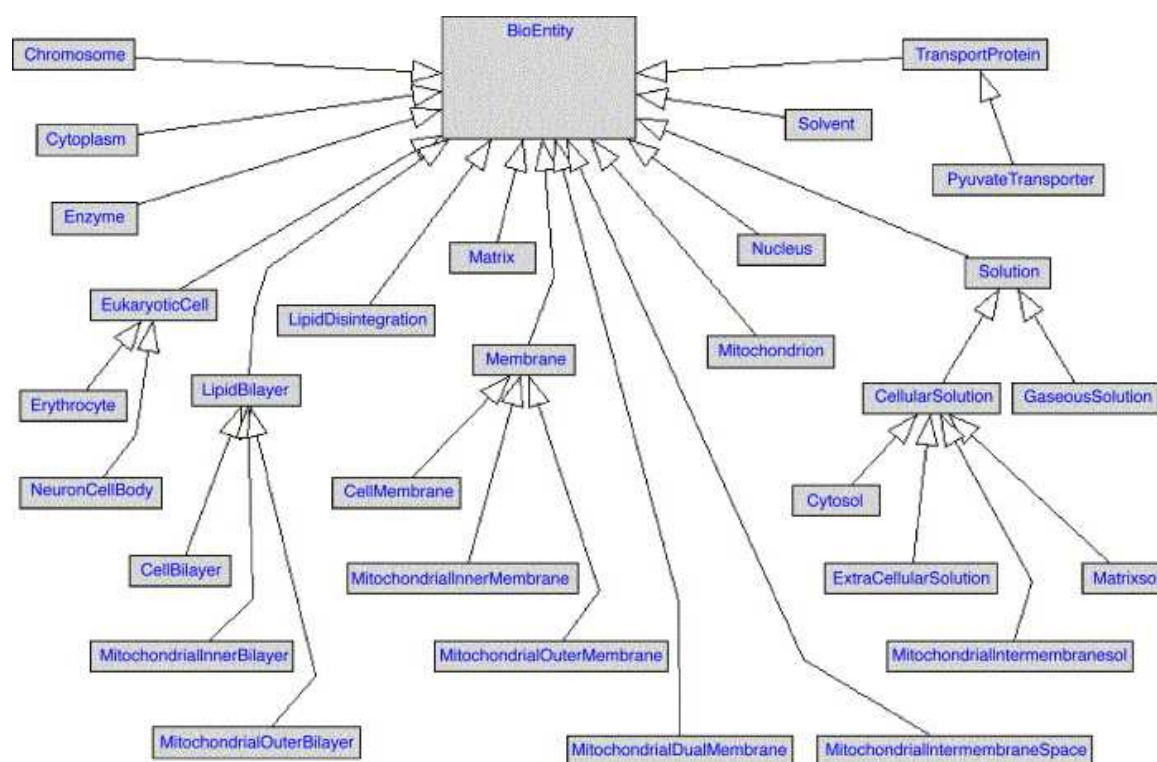


FIG. 2.12: La figure issue de [Webb and White, 2004] représente un diagramme d'héritage de classe de modèle biologique. La notion d'héritage se justifie si on considère les mécanismes naturels d'évolution des espèces et de différenciation moléculaire.

objets constituant le modèle. L'implémentation et l'exécution des programmes doivent permettre l'émergence de la dynamique du système sans passer par un système mathématique global. Ainsi, CellAK permet une description *top-down* des modèles. Dans la même logique, nous pouvons citer la tentative SB-UML [Roux-Rouquié et al., 2004], fournissant un profil spécifiant les éléments standards d'UML pour la biologie systémique. L'article de revue [Johnson et al., 2003] argumente en détail, sur l'intérêt de la POO pour la modélisation intracellulaire.

L'avantage de ces formalisations par rapport aux langages tel que SBML ou CellML, c'est qu'elles permettent d'envisager des composants récopiables. La POO permet de définir des abstractions des modèles biologiques qu'il ne reste plus qu'à instancier pour composer et étudier un modèle plus vaste. Cette vision modulaire doit amener une plus grande souplesse quant à la simulation de systèmes dont la structure est en constante évolution.

Comme toute modélisation, la modélisation par le formalisme d'ingénierie UML requiert quelques précautions. Il peut être inadapté de considérer un système issu de l'évolution naturelle comme un système construit de la même manière qu'un logiciel conçu par l'industrie. Le découpage en classes, les notions d'états discrets, la prédétermination des transitions entre états, peuvent si on n'y prête pas attention aboutir à une modélisation réductrice ou biaisée des systèmes.

UML est un formalisme graphique de la modélisation objet qui ne règle pas, en général, les détails d'implémentation. Les articles proposant UML pour la description des modèles biologiques sont malheureusement assez évasifs quant aux liens entre cette description et le logiciel effectif qui permettrait de simuler. Or, ces liens forts avec l'implémentation doivent

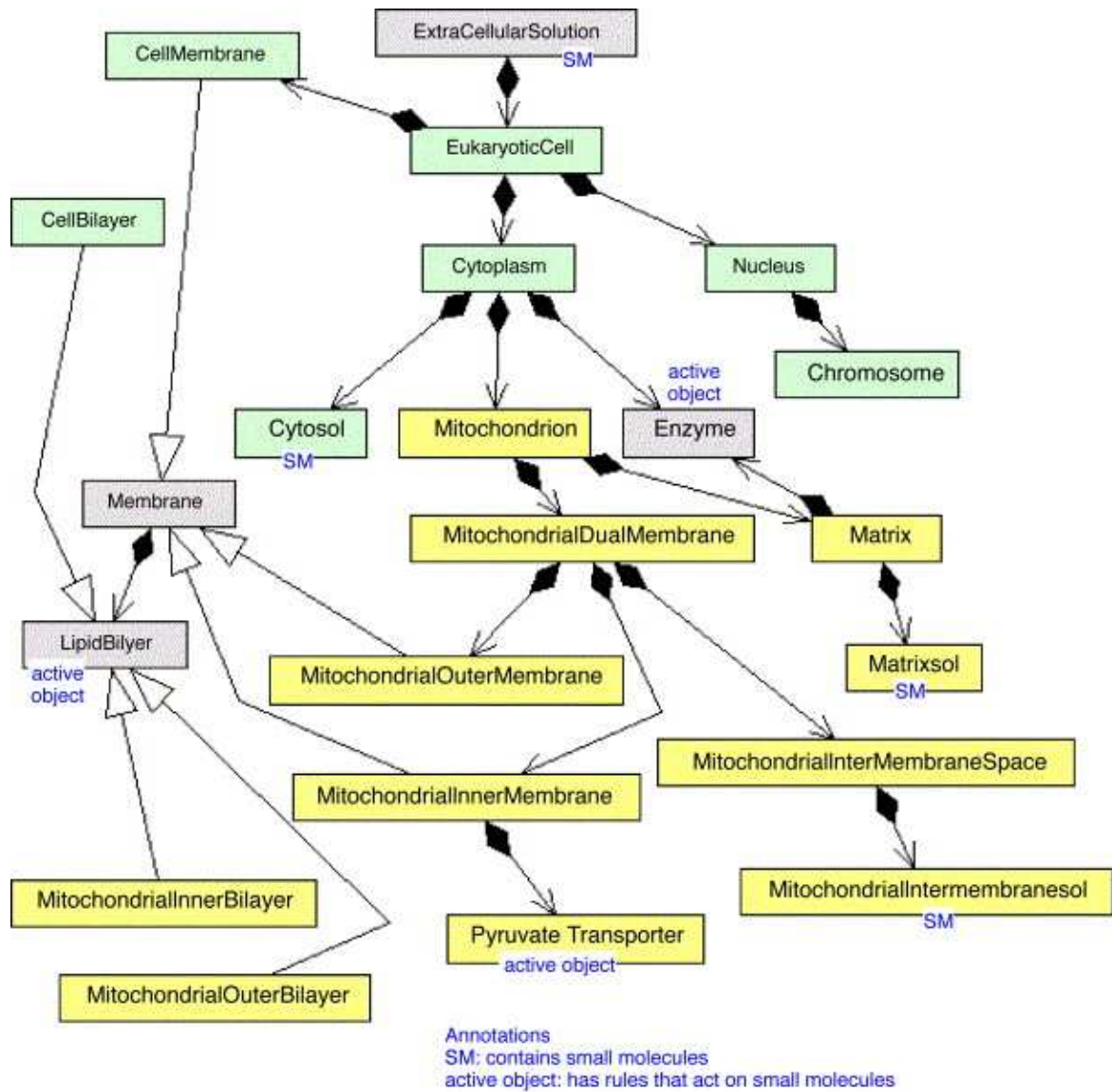


FIG. 2.13: Le schéma issue de [Webb and White, 2004] montre comment on peut définir les liens de composition entre les entités biologiques modélisées.

permettre de faire d'UML quelque chose de plus qu'un formalisme graphique de modélisation en biologie.

Au final, pour utiliser la POO de manière adéquate, il faut allier la prise en compte des paradigmes systémiques et les contraintes de programmation. La tendance semble montrer que la solution s'oriente vers la famille des SMA.

2.4.5 SMA et automates cellulaires

2.4.5.1 Paradigmes individus-centrés

La famille des systèmes multi-agents est vaste. Elle fait partie du domaine de la simulation. Aucune théorie ne permet de déduire *a priori* le comportement de tels systèmes à partir du comportement des composants (les agents). Seule la simulation permet l'étude de ces systèmes et l'émergence d'une explication *a posteriori*. Pour cibler notre propos, nous pouvons identifier trois catégories.

Citons tout d'abord la première catégorie que sont les automates cellulaires. Ils ont été inventés par Stanislaw Ulam et John Von Neumann à la fin des années 40. L'objectif de départ était l'étude de l'auto-reproduction. Dans les années 60, Conway invente le plus célèbre de leurs ancêtres, le « jeu de la vie ». L'histoire des automates cellulaires a continué d'évoluer jusqu'à aujourd'hui. Petit à petit un nouveau paradigme scientifique s'est développé. Ses caractéristiques principales étant de traiter les problèmes selon une approche ascendante (du simple vers le complexe), parallèle et en déterminant les comportements des entités élémentaires de façon locale. Un automate cellulaire est un système dynamique dans lequel le temps et l'espace sont discrets. Il évolue donc sur un réseau maillé (le plus souvent une grille en deux dimensions à mailles carrées). Les automates cellulaires sont synchrones et déterministes. Pourtant, aucune théorie ne permet de prévoir *a priori* leurs comportements complexes et parfois chaotiques.

Ensuite sont apparus les systèmes multi-agents. Les premiers travaux sur les systèmes multi-agents datent de la fin des années 1970 [Hewitt, 1977]. Ils sont motivés par la constatation suivante : « Il existe dans la nature des systèmes capables d'accomplir des tâches collectives complexes dans des environnements dynamiques, sans contrôle externe ni coordination centrale » [Tisseau, 2001]. Les colonies d'insectes sont certainement les exemples les plus notoires d'organisations collectives [Bonabeau et al., 1997]. Les notions de robustesse, d'émergence, d'auto-organisation et d'adaptabilité sont donc sous-jacentes dans les systèmes multi-agents et font même partie de leurs fondements. Ceci fait donc de la méthodologie multi-agents un bon candidat pour la modélisation et la simulation de phénomènes biologiques.

À l'instar du vivant, un agent peut être défini de plusieurs façons. Yves Demazeau définit un système multi-agents comme étant composé de quatre concepts clefs que sont les Agents, l'Environnement, les Interactions et l'Organisation. C'est l'approche *Vowels* ou AEIO [Demazeau, 1995].

Un agent (A) est une entité qui est plongée dans un environnement. Il interagit avec cet environnement et est doté d'une certaine autonomie. Il n'y a donc pas de contrôle global du système. L'agent est l'élément de base d'un système multi-agents. Un agent évolue toujours selon un cycle à trois temps :

- perception : ses capteurs lui fournissent une vision de l'environnement local,
- décision : suivant ses intentions, son état interne et sa perception, l'agent choisit une

action à effectuer,

- action : il modifie l'environnement par son action.

L'environnement (E) peut être divisé en deux parties, l'environnement du système multi-agents et l'environnement de l'agent. Le premier est le support des actions des agents. Le second contient à la fois l'environnement du système multi-agents et les autres agents de l'environnement. Par extension, l'environnement est l'ensemble des conditions extérieures susceptibles d'agir sur le fonctionnement d'un agent.

La notion d'interaction (I) correspond à la mise en relation de deux ou plusieurs agents à travers une série d'actions. Ces actions peuvent être directes ou indirectes, c'est-à-dire que l'interaction peut se produire soit directement entre deux agents, soit par l'intermédiaire d'un autre agent ou de l'environnement. C'est cette dynamique des interactions entre composants qui permet l'émergence de nouvelles fonctionnalités plus complexes.

L'organisation (O) est l'agencement qui se produit entre les agents pour former un système. Les agents vont créer un réseau d'interactions qui va former un tout qui sera capable de résister – dans une certaine mesure – à des perturbations. L'interaction est donc le composant de base de toute organisation, organisation nécessairement dynamique.

Avec l'avènement des sciences cognitives et l'intérêt de l'utilisation des SMA pour les sciences humaines et la modélisation d'agents intentionnels, émotionnels, conversationnels..., le concept d'agent s'est affiné en agent cognitif [Wooldridge and Jennings, 1995], qui correspond à une deuxième catégorie. Les perspectives sociologiques et la coopération entre agents sont alors les centres d'intérêts de la modélisation par SMA cognitifs.

La troisième catégorie, celle des agents réactifs [Ferber, 1997], se situe entre les agents cognitifs et les automates cellulaires. La première voie pour définir un agent réactif est de le considérer comme un agent cognitif simplifié. En effet, les programmes mettant en œuvre un nombre massif d'agents ne peuvent supporter que des agents n'effectuant que des calculs assez simples. Les agents réactifs sont de plus utilisés dans l'étude de l'émergence d'une « intelligence » collective à partir d'un ensemble d'agents dépourvus d'intelligence. La représentation du monde et l'état interne conférant une forte autonomie à l'agent cognitif sont considérablement réduits dans le cas d'un agent réactif.

Enfin, Il est également possible de définir les agents réactifs comme étant des automates cellulaires auxquels ont été ajoutées certaines propriétés tels le mouvement, l'asynchronisme, l'aléatoire... pour accroître les possibilités de modélisation.

2.4.5.2 Applications

Les automates cellulaires et les SMA sont des paradigmes de prédilection pour les études du vaste domaine de la vie artificielle³ [Hewitt, 1977]. Ils sont également bien utilisés pour la simulation microscopique moléculaire [Soula et al., 2005] voir atomique [Amar et al., 2004; Lales et al., 2005], ce niveau de modélisation est toutefois très lié à la physique. Entre ces deux niveaux, se trouvent des travaux allant des modèles intra-cellulaires de réactions chimiques mésoscopiques [Wishart et al., 2004] aux tissus cellulaires [Hunter and Borg, 2003; Wishart et al., 2004]. A ce niveau, il apparaît que les automates cellulaires manquent d'expressivité et doivent être transformés en agents réactifs (par exemple les *Dynamique Cellular Automata* [Wishart et al., 2004] peuvent être vus comme des agents réactifs).

Plusieurs types d'architecture sont disponibles dans la littérature. Les premiers travaux

³Nous ne nous intéressons pas à ce domaine dans l'exposé pour éviter d'élargir encore plus cet état de l'art.

utilisent l'analogie évidente entre la cellule et l'agent. Dans ce cas c'est plutôt la famille des agents cognitifs qui est mise à contribution. Ainsi dans [Santos et al., 2004; Mansury et al., 2002] des agents de type cellule permettent de reproduire le comportement d'une tumeur à partir du comportement individuel de chaque cellule. La phase de décision de chaque agent est ainsi régie par une machine à états ou un réseau bayésien. Les différents agents s'envoient des messages pour modéliser les relations inter-cellulaires. L'environnement est également modélisé. Le projet *epitheliome*[Walker et al., 2004], dans la même logique, modélise et simule un tissu d'agents cellules. Le comportement est régi cette fois par un système de règles⁴ modélisant la mitose, l'apoptose, la migration, l'échange de substrat avec l'environnement, les communications inter et intra-cellulaires. Finalement les systèmes sont vus comme des sociétés de cellules.

Viennent ensuite les applications qui visent plutôt à modéliser les phénomènes chimiques intra-cellulaires [Wishart et al., 2004; Webb and White, 2004]. Ce sont les agents réactifs qui sont utilisés à cette échelle, pour simuler un grand nombre de molécules et faire émerger le comportement global de la cellule à partir de phénomènes simples. Une molécule peut être considérée comme une particule, un disque, une sphère... en mouvement dans un espace en 2 ou 3 dimensions, ou sur une grille.

Nous pouvons également citer le simulateur SimBioDyn [Ballet et al., 2004] qui permet la mise au point rapide de modèles biologiques dont les agents géométriques sont liés par des systèmes masses-ressorts. Cette application permet de représenter des phénomènes plutôt d'ordre mécanique telle que la phagocytose. Le projet CyberCell [Broderick et al., 2004] permet de simuler l'évolution spatiale de la membrane d'une cellule à partir des comportements d'agents spatialisés représentant des molécules (par exemple une paire de phospholipides). Le logiciel 3DSpi [Soula et al., 2005] permet d'observer la formation d'agrégation moléculaire et la formation de structures organisées analogues à celles qui peuvent se trouver dans le noyau d'une cellule. Ces trois dernières applications partent finalement de considérations physiques pour remonter vers des faits biologiques.

Enfin, il existe des travaux qui s'orientent vers une méthode générique de modélisation multi-modèles. Les travaux décrits dans [Bosse et al., 2006] utilisent la notion d'organisation grâce à l'approche AGR (*Agent, Group, Role*) hérité de [Ferber and Gutknecht, 1998]. L'exemple d'*E.Coli* est décrit par son organisation, une organisation étant définie par des groupes, des rôles (les agents) et les relations inter-rôles et inter-groupes.

L'architecture en *blackboard* utilisée pour les *cellulat* [González et al., 2003] se rapproche le plus de ce que nous proposerons. L'architecture en *blackboard* ou tableau noir est un concept d'échange d'information qui date du tout début des années 80 [Erman et al., 1980]. Le tableau noir est une zone de travail commune, dévolue à la transition d'informations entre les différents agents. La figure 2.14 illustre l'architecture d'un modèle de cellule utilisant cette méthode.

En conclusion sur l'intérêt de l'approche par SMA, nous constatons un certain nombre d'avantages. D'abord, ils sont adaptés pour faire le lien entre les niveaux microscopique et macroscopique par l'explication *a posteriori* fournie par la simulation. En ce sens, ces applications permettent une observation multi-échelles.

Nous pouvons tout de même déplorer que dans la plupart des applications, la modélisation

⁴Des travaux qui ne sont pas encore publiés semblent indiquer que le projet *epitheliome* tend désormais à utiliser le formalisme des *X-machines*. Pour plus de détails voir : <http://epi1.shef.ac.uk/>

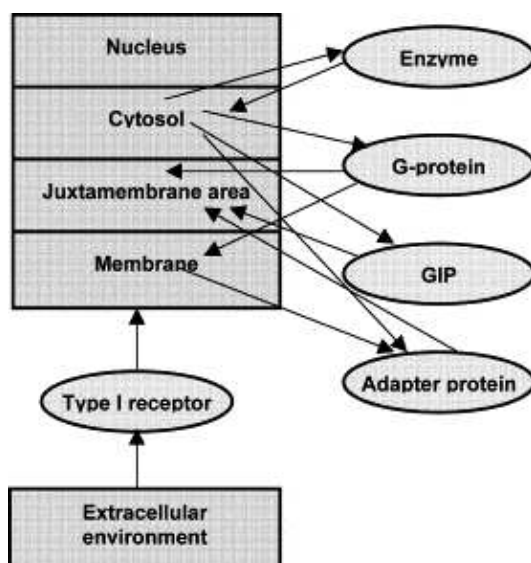


FIG. 2.14: Le schéma issu de [González et al., 2003] montre l'architecture en *cellulas* (ovales)/ *blackboard* (rectangles) d'un modèle de voie de transduction.

se base sur un seul niveau de granularité. La construction classique n'est donc pas vraiment multi-échelles. Cependant, certains travaux que nous venons d'évoquer, s'intéressent plus aux aspects méta modèles SMA et vont dans la direction du multi échelle.

Ensuite, comme les travaux autour de l'application 3DSpi nous l'indiquent [Soula et al., 2005], les phénomènes vivants sont dynamiques, complexes, modulaires, dégénérés⁵ et stochastiques. L'usage des SMA permet de conserver ces propriétés dans les modèles.

Enfin, nous remarquons que les modélisations par objets et agents dans le domaine de la biologie systémique sont récentes et en plein essor. Encore peu d'articles sont publiés dans les journaux. Nous pouvons supposer qu'elles ont un avenir prometteur pour la recherche en biologie inscrite dans l'ère post-génomique.

2.5 Conclusion

Dans le premier chapitre de cet état de l'art, nous avons fait état d'un changement de paradigme en biologie, provoqué par la prise en compte de la complexité du vivant.

Dans ce chapitre, nous avons montré comment l'informatique peut être mise à contribution pour étudier cette complexité en donnant naissance aux calculs *in silico* par analogie avec les différents moyens classiques d'expérimentation. Nous avons également constaté que la simulation joue un rôle essentiel en ce domaine. La prise en compte des aspects multi-échelles et multi-modèles est le défi actuel de la simulation.

Ensuite, il apparaît que le cycle classique de simulation est celui utilisé par *Virtual Cell* : établissement du modèle biologique, mathématisation du problème, résolution, interprétation. Les outils plus orientés objet permettent de s'affranchir de la phase de mathématisation en distribuant les éléments de résolution numérique dans les composants du modèle. L'utilisation

⁵Des structures différentes peuvent remplir une même fonction et, inversement, une même structure peut contribuer à des fonctions différentes. Il est fort probable que l'apparente spécificité des voies métaboliques et/ou de signalisation est tout autant révélatrice d'une démarche méthodologique que d'une réalité biologique.

des SMA semble être le moyen technique de mettre en œuvre ce type de modélisation. Un des avantages de ces méthodes est de permettre la visualisation de la simulation par d'autres moyens que la simple observation de courbes. Ce faisant elle permet l'explication *a posteriori* beaucoup plus efficacement, à tel point que la visualisation apparaît comme une possibilité incontournable d'une application visant à l'étude des phénomènes biologiques [Dorin, 2002].

Finalement remarquons que l'article « *Computational systems biology* » [Kitano, 2002] fait l'usage de l'expression « *in silico experiments* ». Cette idée est récurrente dans bon nombre d'articles. Elle mérite d'être développée et précisée, ne serait-ce pour consolider l'intérêt de la simulation qui ne permet pas la même analyse *a priori* que les méthodes formelles « biologiques ».

Chapitre 3

Complexité biologique, informatique et cognition

Nous voici maintenant à nous interroger sur la notion d'expérience. Au chapitre précédent, nous avons constaté que les calculs *in silico* fournissent un outil d'investigation complétant les expériences *in vivo* et *in vitro*. Ils permettent d'avancer dans notre compréhension et notre connaissance des phénomènes biologiques. Ces travaux *in silico*, peuvent servir à démontrer formellement des propriétés ou encore à observer des résultats issus de la simulation. Nous l'avons vu, dans ce dernier cas de figure, certains auteurs parlent d'expériences *in silico*.

Cette notion d'expérience nous intéresse grandement. En premier lieu, les sciences cognitives apporteront quelques indications quant à cette notion et à son rôle dans la production de connaissances. Ensuite, nous montrerons que la réalité virtuelle est construite autour de cette idée d'expérience et qu'elle est adaptée à l'étude des systèmes vivants. Pour se faire nous répondrons aux trois questions : *quels concepts ? quels outils ? et quels modèles ?*

3.1 Cognition, expérience et science

En préambule de ce chapitre, cette section présente une vision des sciences et techniques de la cognition de laquelle nous déduisons différents degrés d'expérience pour la démarche scientifique.

Le premier chapitre fut consacré à l'étude de l'émergence d'un nouveau paradigme scientifique de la complexité en biologie. Cette transition s'opère à l'instar de ce qui a pu se passer auparavant en sciences physiques. Ensuite, nous avons montré comment l'outil informatique essaie de rendre compte de cette complexité biologique. Intéressons nous également au fait que cette notion de complexité joue également un rôle quant à la compréhension de la « connaissance » avec l'émergence des sciences cognitives. En effet, nous devons nous intéresser aux sciences cognitives parce qu'au final ce qui est visé avec l'outil *in silico*, c'est la connaissance, et parce que les mêmes idées qui dirigent la construction de nos modèles ont un impact sur notre conception de la connaissance. Nous pourrions alors mieux comprendre l'expérience et son intérêt pour la connaissance.

3.1.1 La cognition science de la complexité

Pour comprendre l'histoire des sciences cognitives, nous nous basons principalement sur la vision de Varela [Varela, 1989b; Varela et al., 1993]. L'histoire de cette science peut être découpée en quatre étapes : le béhaviorisme, le computationnalisme, le connexionnisme et le constructivisme.

Tout d'abord, nous pouvons considérer la non prise en compte de la cognition comme une vision béhavioriste. Le béhaviorisme ou comportementalisme peut être décrit comme une approche qui se concentre sur l'étude du comportement observable et du rôle de l'environnement en tant que déterminant du comportement. Un individu est alors considéré comme une boîte noire réactive, produisant une réponse à un stimulus.

Les sciences et techniques de la cognition (STC), issues des contributions de diverses disciplines telles les neurosciences, la linguistique, la psychologie cognitive, l'épistémologie ou encore l'intelligence artificielle, tentent d'explicitier cette boîte noire.

Les STC ont également hérité de l'époque des conférences de Macy (1946-1953) durant lesquelles John von Neumann, Robert Wiener, Warren Mc Culloch et d'autres scientifiques venant de disciplines variées, voulaient définir une science de l'esprit. Wiener la nomma *cybernetic* [Wiener, 1952]. Les retombées scientifiques de cette période sont innombrables dans différents domaines scientifiques. La recherche et la définition de principes généraux gouvernant les systèmes complexes font de la cybernétique un des ancêtres du paradigme systémique. Ensuite, l'invention de l'architecture de l'ordinateur de Von Neumann, encore utilisée aujourd'hui, donne naissance à l'informatique. Elle conditionne de plus l'orientation des premiers cognitivistes. Ainsi, dans les années 50, Herbert Simon, Noam Chomsky, Marvin Minsky et John Mc Carthy définissent la cognition par la *computation* de représentations symboliques.

Le programme de recherche du computationnalisme devient alors le fondement de la branche principale des STC que l'on appelle le cognitivisme. Le cognitivisme peut être défini par les réponses aux questions suivantes [Varela, 1989b] :

Question 1 : Qu'est-ce que la cognition ?

Réponse : Le traitement de l'information : la manipulation de symboles à partir de règles.

Question 2 : Comment cela fonctionne-t-il ?

Réponse : Par n'importe quel dispositif pouvant représenter et manipuler des éléments physiques discontinus : des symboles. Le système n'interagit qu'avec la forme des symboles (leurs attributs physiques) et non leur sens.

Question 3 : Comment savoir qu'un système cognitif fonctionne de manière appropriée ?

Réponse : Quand les symboles représentent adéquatement quelques aspects du monde réel, et que le traitement de l'information aboutit à une solution efficace du problème soumis au système.

Cette vision du cognitivisme domine largement les STC depuis quarante ans. Varela la nomme : « l'orthodoxie cognitiviste ». L'intelligence artificielle en est bien souvent la représentante dans le domaine des « sciences de l'information » avec les systèmes experts, le langage Prolog (PROgrammation LOGique), les techniques du traitement de l'image...

Le fait qu'on ne trouve pas de règles, ni de processeur logique dans le cerveau humain et que l'information n'y est pas stockée à une adresse précise, constitue une remise en cause de l'orientation cognitiviste. Le connexionnisme présente l'émergence comme une alternative à la représentation symbolique critiquée du cognitivisme. Les réseaux de neurones artificiels

sont peut-être l'outil de l'intelligence artificielle le plus représentatif de ce mouvement.

La stratégie du connexionnisme est de construire un système cognitif à partir, non pas de symboles et de règles, mais à partir de constituants simples qui peuvent être dynamiquement reliés les uns aux autres. Ainsi une coopération globale peut émerger¹ spontanément, bien que chaque constituant fonctionne dans un environnement local.

On peut définir cette tendance des STC par les réponses suivantes :

Question 1 : Qu'est-ce que la cognition ?

Réponse : L'émergence d'états globaux dans un réseau de composants simples.

Question 2 : Comment cela fonctionne-t-il ?

Réponse : Des règles locales gèrent les opérations individuelles et des règles de changement gèrent les liens entre les éléments.

Question 3 : Comment savoir qu'un système cognitif fonctionne de manière appropriée ?

Réponse : Quand les propriétés émergentes (et la structure résultante) sont identifiables à une faculté cognitive - une solution adéquate pour une tâche donnée.

La modélisation connexionniste est plus compatible avec la réalité physiologique du cerveau. Le sens n'est pas enfermé à l'intérieur de symboles : il est fonction de schémas d'activités complexes émergeant d'une interaction entre plusieurs d'entre eux. Finalement on peut parler d'émergence symbolique. Nous voyons que ce qui sépare théoriquement le cognitivisme du connexionnisme, est principalement un problème de niveau représentationnel. Il est possible de voir le cognitivisme comme une méthode descendante et le connexionnisme comme une méthode ascendante.

Les travaux de Varela remettent en cause la notion même de représentation dans le cadre de la cognition. L'approche énaïve ou « énaïon », propose une conception incarnée de la cognition définie comme « faire-émerger un monde issu d'une histoire de couplage structurel », compatible avec l'ensemble des tendances des STC. On peut ainsi définir l'énaïon par les réponses suivantes :

Question 1 : Qu'est-ce que la cognition ?

Réponse : L'action productive : l'historique du couplage structurel qui énaïe (fait émerger) un monde.

Question 2 : Comment cela fonctionne-t-il ?

Réponse : Par l'entremise d'un réseau d'éléments inter-connectés, capable de subir des changements structuraux au cours d'un historique non interrompu.

Question 3 : Comment savoir qu'un système cognitif fonctionne de manière appropriée ?

Réponse : Quand il s'adjoit un monde de significations préexistant, en continuel développement (comme c'est le cas des petits de toutes les espèces), ou qu'il en forme un nouveau (comme cela arrive dans l'histoire de l'évolution).

L'énaïon insiste sur la circularité totale entre la représentation et l'action. C'est l'action/interprétation qui permet de faire émerger un monde. L'originalité de cette approche réside dans la prédominance de l'action sur celui de la représentation. En d'autres termes, le modèle de l'énaïon nous indique que l'expérience est indispensable à la connaissance.

Plus généralement, il apparaît aujourd'hui que l'énaïon est un modèle particulier d'un mouvement des STC appelé constructivisme. La mise en avant de l'expérience est quelque chose de très présent dans le constructivisme. Plus on expérimente un système, plus on en construit sa connaissance, plus on le comprend. Nous avons choisi de nous intéresser au modèle particulier de l'énaïon pour plusieurs raisons.

¹Plusieurs termes peuvent être utilisés : auto-organisation, émergence de propriétés ou encore synergétique.

En premier lieu, il semble que ces idées soient actuellement en plein essor² dans la communauté des psychologues.

Ensuite, nous le verrons, cette conception de la cognition est compatible avec l'utilisation de la réalité virtuelle. Pour préciser cela nous détaillons dans cette section, ci-après, ce que nous comprenons en pratique, par « expérience ».

Enfin, l'énaction a pour origine les travaux sur la compréhension du vivant menés par Varela. Ces travaux, sur lesquels nous reviendrons, définissent l'autonomie, l'autopoïèse, l'homéostasie, la clôture opérationnelle... Ils sont la source d'inspiration de ce que nous proposons dans cette thèse pour modéliser le vivant. Nous nous servons de ces deux volets des travaux de Varela sur deux plans différents : ici, pour l'introduction du concept de réalité virtuelle et au chapitre suivant pour la conception d'un modèle pour la biologie. Ces deux aspects sont liés dans leur justification mais n'ont *a priori* rien à voir dans leur application. Toutefois, il est appréciable de manipuler à tous les niveaux des idées compatibles entre elles.

3.1.2 Le rôle de l'expérience

L'expérience est le point de départ de toute science et de toute théorie. C'est déjà ce qu'affirmait la phénoménologie avant l'apparition des sciences cognitives et du mouvement constructiviste. Si la notion d'expérience est fondamentale en science, elle l'est d'autant plus en biologie que l'on nomme : science expérimentale (voir section 1.1). Pouvons-nous alors la définir plus précisément ?

Il est possible de distinguer plusieurs degrés dans les notions d'expérience et de science expérimentale. Claude Bernard[Bernard, 1865] opérait une distinction entre : les sciences de l'observation qui décrivent les phénomènes et permettent de les prévoir ; et les sciences expérimentales qui non seulement permettent de prévoir les phénomènes mais également d'agir sur eux. Le niveau zéro de l'expérience, c'est donc l'observation. Ensuite vient l'aspect expérimental qui s'étend de l'*expérimentation* à l'*expérience*.

Par *expérience*, nous entendons « faire l'expérience avec son corps » comme le conçoit l'énaction. L'action, la perception et l'interprétation sont trois activités indissociables. Les conséquences des actes sont alors vécues instantanément.

Par *expérimentation*, nous faisons référence à l'acte qui consiste à observer l'état d'un modèle après l'application d'un stimulus. L'*expérimentation* est ainsi une « *expérience* en temps différé ». Elle est plus indirecte et se faisant, elle dissocie l'action et la perception ; ce qui la rend bien moins efficace d'un point de vue cognitif. Nous comprenons moins bien ce que nous observons que ce que nous vivons.

Bien entendu, l'*expérimentation* est souvent le seul recours pour étudier des phénomènes qui se réalisent à des échelles de temps ou d'espace auxquelles notre corps ne peut accéder directement. De plus, la frontière entre ces deux extrêmes n'est pas clairement définie. Dans les faits, la pratique expérimentale se situe quelque-part entre les deux. D'ailleurs, le sens des termes *expérimentation* et *expérience* est bien plus large dans son acception commune que dans la définition que nous donnons ici, pour les besoins de notre argumentation.

Pour résumer, nous considérons maintenant trois niveaux dans l'expérience³ :

²Il se crée de plus en plus de colloques, écoles d'été, conférences... autour de l'idée d'énaction. Citons par exemple, l'écolde thématique CNRS : « *Constructivisme et éniation : Un nouveau paradigme pour les sciences cognitives* » (<http://liris.cnrs.fr/enaction/>), à Oléron, en 2006.

³À titre d'exemple, prenons le cas d'un élève apprenant l'informatique. L'observation c'est le cours magistral. L'expérimentation c'est l'exercice de travaux dirigés, que l'on corrige au tableau. L'expérience c'est les travaux pratiques sur la machine. En ce domaine, l'intérêt pédagogique des travaux pratiques n'est plus à

- 0 - L'observation : la perception seule ;
- 1 - L'*expérimentation* : l'action puis la perception en temps différé ;
- 2 - L'*expérience* : l'action et la perception indissociées (l'énaction).

Au chapitre précédent (voir section 2.3), nous avons parlé des calculs *in silico*. Les utilisateurs de ces outils se réclament de la démarche expérimentale. D'après notre échelle de valeur, nous pouvons en effet parler d'*expérimentation in silico*, le niveau 1 de l'expérience. Souvent, les biologistes se contentent de ce niveau tant les modèles *in vivo* et *in vitro* expérimentés sont difficiles à manipuler (lorsqu'on s'intéresse au niveau cellulaire par exemple). Par contre, cette limitation se justifie moins dans le cas de la manipulation de modèles numériques. Toutefois, on pourrait l'expliquer par la méthode classique de simulation des modèles qui consiste à mathématiser le modèle. En effet, cette mathématisation fait disparaître le sens concret du sujet de l'étude (sens physique ou sens biologique). Ensuite le système mathématique est résolu et les résultats sont réincarnés. Cette absence d'ontologie pendant la phase mathématique ne permet pas l'*expérience*, puisqu'il n'y aurait plus de sens à agir à ce moment, sur un modèle désincarné.

Nous pensons qu'il est possible de proposer des outils qui permettent de dépasser le niveau l'*expérimentation in silico* pour permettre véritablement l'expérience des modèles numériques. Nous le détaillerons, la réalité virtuelle est dans cet objectif, l'outil idéal.

3.2 Réalité virtuelle et biologie : Quels concepts ?

La réalité virtuelle entretient des liens étroits avec les notions de simulation, d'expérience et de cognition. Cela en fait un domaine privilégié pour appréhender les systèmes complexes et par conséquent, un candidat à l'étude des phénomènes biologiques. Il convient alors d'exposer clairement les concepts de la réalité virtuelle.

Dans cette section, nous commençons par nous attarder sur l'appellation ambiguë de cette branche de l'informatique. Nous la définissons ensuite, et décrivons l'outil qui en découle : l'*expérimentation in virtuo*. Nous constatons qu'il s'agit là d'un outil de choix pour la biologie et les sciences interdisciplinaires.

3.2.1 Étymologie

Construite sur l'opposition du sens courant de ses deux composantes, la formule réalités virtuelles est absurde. Qui vous parlerait de la couleur blanc-noir vous apparaîtrait comme un esprit confus ne sachant pas lui-même, en associant un mot et son contraire, de quoi il veut parler. Certes, en toute rigueur, virtuel et réel ne sont pas en opposition. Le virtuel, du latin virtus (vertu, force), est ce qui est en puissance dans le réel, ce qui a en soi toutes les conditions essentielles à sa réalisation ; mais alors, que peut bien être une réalité qui a en soi toutes les conditions de sa réalisation ? Vue sous cet angle, la formule est encore plus inepte. C. Cadoz, Les réalités virtuelles [Cadoz, 1994].

L'expression anglo-saxonne *virtual reality* est proposée pour la première fois en juillet 1989, lors d'un salon professionnel⁴, par Jaron Lanier, alors responsable de la société VPL

démontrer.

⁴Texpo'89 à San Francisco (USA)

Research, spécialisée dans les périphériques d'immersion. Il forge cette expression dans le cadre de la stratégie marketing et publicitaire de son entreprise, sans chercher à en donner une définition très précise. En anglais, selon le BBC English Dictionary (HaperCollins Publishers, 1992), *Virtual : means that something is so nearly true that for most purposes it can be regarded as true, also means that something has all the effects and consequences of a particular thing, but is not officially recognized as being that thing*. Ainsi, une *virtual reality* est une quasi-réalité qui a l'apparence et le comportement d'une réalité mais qui n'en est pas une : il s'agit plutôt d'un ersatz ou d'un succédané de réalité. La traduction littérale de l'expression anglo-saxonne donne en français le terme réalité virtuelle, qui devient alors une expression absurde et inepte, comme le souligne la citation placée en exergue de cette section. En effet, selon Le Petit Robert (Éditions Le Robert, 1992), *Virtual : [qui] a en soi toutes les conditions essentielles à sa réalisation*. Une réalité virtuelle serait donc une réalité qui aurait en soi toutes les conditions essentielles à sa réalisation ; ce qui est bien le moins pour une réalité ! Ainsi, de l'anglais au français, le terme de réalité virtuelle est devenu équivoque. Il relève du procédé rhétorique appelé oxymoron, qui consiste à réunir deux mots qui semblent contradictoires. Ce type de construction donne à l'expression un caractère inattendu qui, nous devons bien en convenir, est plus médiatique que scientifique. L'usage de l'adjectif « vicariant⁵ » aurait été plus exact quant à la traduction de *virtual* [Fuchs et al., 2001]. D'autres expressions telles que cyber-espace [Gibson, 1980], réalité artificielle [Krueger, 1983], environnement virtuel [Ellis, 1991], monde virtuel [Holloway et al., 1992], sont également proposées, mais l'antonymie réalité virtuelle reste encore très largement utilisée comme le montre une rapide recherche sur la Toile. Un autre point de vue consiste à considérer le réel comme ce qui existe par soi-même indépendamment du fait que nous puissions le percevoir ou le concevoir (Dictionnaire historique de la langue française, Robert, 2000). Une réalité est alors une représentation du réel (un modèle) et une réalité virtuelle serait donc une représentation virtuelle, ce qui est bien le moins pour un modèle ! L'expression réalité virtuelle devient alors un pléonasme [Mellet d'Huart, 2004].

Entre oxymoron et pléonasme, ces équivoques entretiennent la confusion. Heureusement la communauté des chercheurs en réalité virtuelle la définit assez rigoureusement.

3.2.2 Définitions

La première définition est une définition théorique fonctionnelle donnée par P. Fuchs dans le traité de la réalité virtuelle [Fuchs et al., 2003] : *La réalité virtuelle permet à l'utilisateur de s'extraire de la réalité physique pour changer virtuellement de temps, de lieu et(ou) de type d'interaction : interaction avec un environnement simulant la réalité ou interaction avec un monde imaginaire ou symbolique*. Cette définition est complétée par une définition technique plus concrète : *les techniques de la réalité virtuelle sont fondées sur l'interaction en temps réel avec un monde virtuel, à l'aide d'interfaces comportementales permettant l'immersion « pseudo naturelle » de l'utilisateur dans cet environnement*.

Pour relever de la réalité virtuelle (RV) une application informatique doit donc posséder les attributs/propriétés suivantes :

Un monde virtuel : C'est la représentation d'une réalité : le modèle d'étude.

Le temps réel : L'écoulement du temps virtuel doit être maîtrisé (*i.e.* équivalent ou

⁵D'après le Petit Robert (Éditions Le Robert, 1992) : *Qui remplace, qui se substitue à autre chose*. En biologie on parle également du rôle vicariant d'un organe capable de suppléer à l'insuffisance fonctionnelle d'un autre organe.

tout au moins proportionnel au temps réel). Par exemple, dix secondes du temps réel (celui de notre montre) peuvent correspondre à une seconde de temps virtuel (celui du monde virtuel).

L'immersion : L'application doit exercer une action permettant à l'utilisateur de percevoir (avec ses sens) le monde dans lequel il se trouve. Nous parlons alors de médiation des sens. Un écran est bien sûr l'interface la plus simple qui vient à l'esprit.

L'interaction : À tout moment, l'application doit permettre à l'utilisateur d'agir sur le monde virtuel. Nous parlons de médiation de l'action. Le temps de latence concernant la prise en compte d'une action doit bien sûr être minimisé et adapté à l'application considérée. La souris et le clavier sont ici les outils d'interface les plus communs.

Finalement, ces deux définitions permettent de définir assez largement le domaine et ce qui relève de la RV. Face à une application de RV l'utilisateur subit ainsi l'action d'une certaine réalité. Il perçoit également les effets de ses actions sur l'action qu'a cette réalité sur lui. Il fait ainsi l'expérience de cette réalité en réalisant l'énaction. Nous avons alors relevé deux types de médiation, celle des sens et celle de l'action. Remarquons que ces deux médiations indissociables dans le concept d'énaction, impliquent malgré tout deux propriétés distinctes pour un programme informatique. Ces propriétés satisfaites, la RV permet alors un nouveau mode d'investigation des modèles numériques : l'expérience *in virtuo*.

3.2.3 L'expérience *in virtuo*...

L'expression *in virtuo* (dans le virtuel) est un néologisme construit par analogie avec les locutions *in vivo*, *in vitro* et *in silico*. Nous avons déjà présenté l'usage de l'expression *in silico* (voir section 2.3) pour qualifier les calculs sur ordinateur ; cependant, *in silico* n'évoque pas la participation de l'homme à l'univers de modèles numériques en cours d'exécution : c'est pourquoi nous lui préférons *in virtuo* qui, par sa racine commune, rappelle les conditions expérimentales de la réalité virtuelle [Tisseau, 2001].

Dépassant la simple observation de l'activité du modèle numérique en cours d'exécution sur un ordinateur, l'utilisateur peut tester la réactivité et l'adaptabilité du modèle en fonctionnement, tirant ainsi profit du caractère comportemental des modèles numériques. Ce type d'expérimentation/expérience est appelé : l'expérimentation *in virtuo*. Elle permet donc de pratiquer pleinement l'*expérience*, telle que nous l'avons définie (voir section 3.1.2).

Une expérimentation *in virtuo* est ainsi une expérimentation conduite dans un univers virtuel de modèles numériques en interaction et auquel l'homme participe. La réalité virtuelle implique pleinement l'utilisateur dans la simulation, rejoignant ainsi l'approche de la conception participative qui préfère voir dans les utilisateurs des acteurs humains plutôt que des facteurs humains. Une telle simulation participative en réalité virtuelle met en œuvre des modèles de types différents (multi-modèles) issus de domaines d'expertise différents (multi-disciplines). Elle est souvent complexe car son comportement global dépend autant du comportement des modèles eux-mêmes que des interactions entre modèles. Enfin, elle doit inclure le libre arbitre de l'utilisateur humain qui exploite les modèles en ligne.

L'expérimentation *in virtuo* implique ainsi un vécu que ne suggère pas la simple analyse de résultats numériques. Entre les preuves formelles *a priori* et les validations *a posteriori*, se place une réalité virtuelle vécue par l'utilisateur qui peut ainsi franchir le cap des idées reçues pour accéder à celui des idées vécues. La figure 3.1 replace l'expérimentation *in virtuo* dans le contexte de la modélisation.

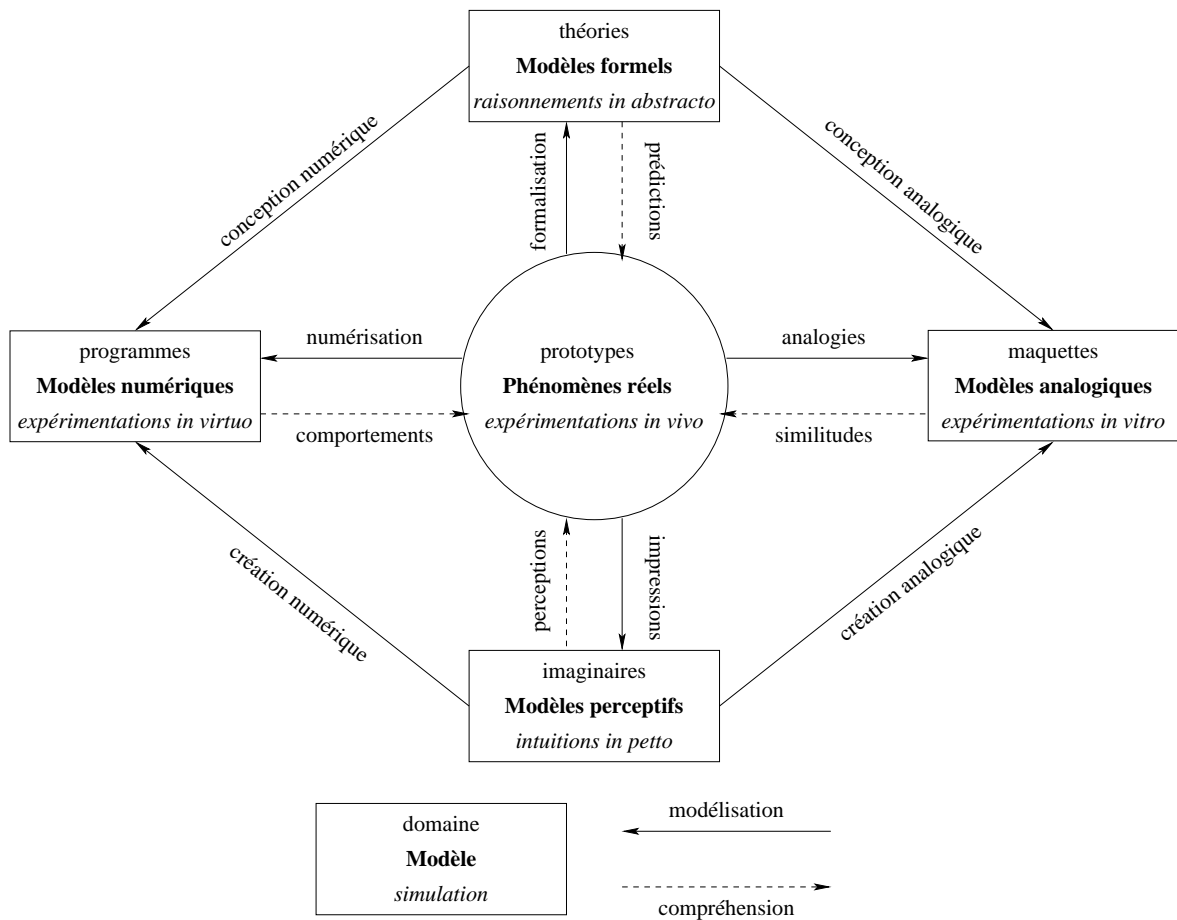


FIG. 3.1: Modélisation, simulation et compréhension des phénomènes (issue du *manifeste du CERV*).

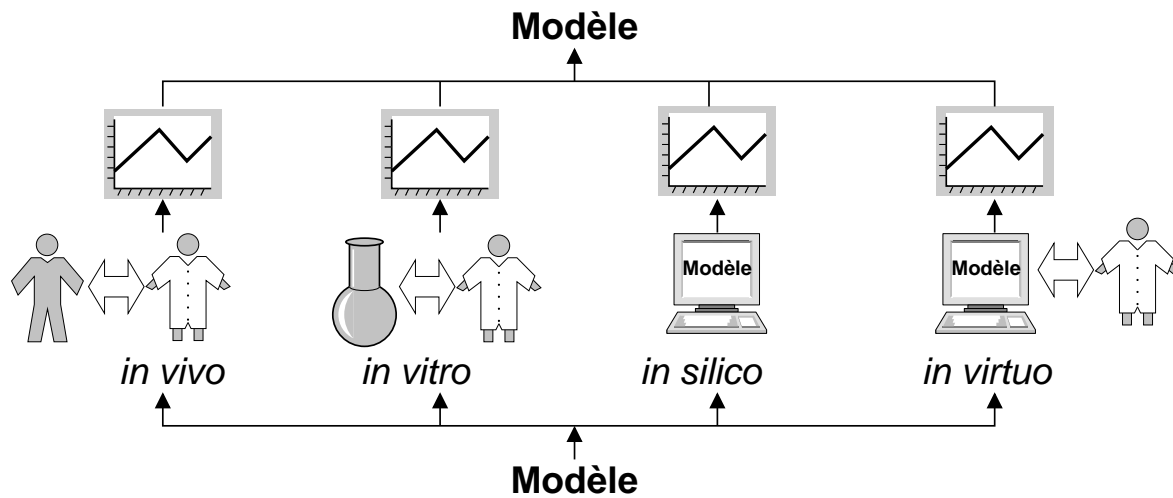


FIG. 3.2: Les différents modes d'investigation du vivant.

3.2.4 ...pour la biologie.

L'expérience *in virtuo* nous apparaît alors comme un outil de choix pour l'étude des modèles du vivant (et par extension pour l'étude du vivant lui même). Citons alors quelques possibilités, certes déjà suggérées par la « simple » simulation, mais renforcées par ce nouvel outil élevant le niveau d'expérience autorisé.

D'abord, l'expérimentation *in virtuo* favorise l'acte de modélisation. Son champ d'application multi-modèles permet de rassembler une grande partie de l'état des connaissances sur un sujet particulier, au sein d'un même modèle. Se faisant, le modélisateur transforme un savoir souvent construit comme une apposition de connaissances isolées, en un ensemble articulé. Cela permet notamment de mettre à jour les incohérences des connaissances bibliographiques en biologie. Il est à noter, nous le verrons par la suite, que les actes de modélisation et d'expérimentation ne sont pas nécessairement dissociés (grâce à la médiation de l'esprit, voir section 3.3.2.2)

Ensuite, la manipulation de modèles théoriques matérialisés dans l'univers virtuel possède des très bonnes vertus pédagogiques. En cela, elle favorise la collaboration entre chercheurs et donc le travail interdisciplinaire. Nous pouvons décrire deux niveaux de collaboration pour lequel le modèle *in virtuo* est un bon média. Le premier niveau permet l'échange et la collaboration entre modélisateurs spécialistes des divers domaines cloisonnés de la biologie. Le second permet l'échange entre les biologistes théoriciens et ceux plus habitués à la paillasse. Par exemple, les scientifiques rompus aux techniques de la biologie cellulaire ne possèdent pas suffisamment d'expertise en mathématiques, en chimie, en physique ou en informatique pour générer des prédictions quantitatives à partir des modèles. Les biologistes théoriciens ou systémiciens sont eux plus souvent habitués aux sciences physiques, mais ont du mal à communiquer avec les expérimentateurs (par exemple, un diagramme de bifurcation ne peut pas servir de base au langage commun).

Enfin, en plus de permettre l'énaction du modèle numérique représentant l'état des connaissances d'un phénomène, l'expérience *in virtuo* permet la formulation de nouvelles hypothèses, de nouvelles pistes pour l'expérimentation sur le système réel. Car, les modèles des systèmes biologiques complexes sont eux aussi suffisamment complexes pour mériter d'être étudiés par cette démarche qu'est l'expérience. Il nous reste cependant à définir quels outils

permettent l'expérience *in virtuo*.

3.3 Le virtuoscope : Quels outils ?

3.3.1 Les travaux du Centre Européen de Réalité

Le Centre Européen de Réalité Virtuelle⁶ situé à Brest, a pour objet l'utilisation de la réalité virtuelle pour l'étude des systèmes complexes. Ses activités sont organisées en quatre équipes, qui tissent chacune selon un point de vue différent, les relations entre la réalité virtuelle et la modélisation des systèmes complexes :

- **ASAP** (Apprentissages, Stratégies, Attitudes, Performances) : L'équipe ASAP se situe dans le cadre de la psychologie des apprentissages. Elle a pour objectif d'étudier les effets des modifications volontaires des attitudes (pré-perceptions, postures, préjugés) sur les performances intellectuelles ou motrices par le biais d'une stratégie de gestion posturale et/ou d'une stratégie d'imagerie mentale. Plus précisément, il s'agit d'identifier dans quelle mesure un travail systématique sur les attitudes, sollicité par de l'imagerie mentale ou virtuelle, facilite l'apprentissage de compétences nouvelles et/ou l'amélioration de performances.
- **SPI** (Simulation Participative et Immersive) : Le projet SPI a pour objectif de simuler des systèmes complexes intégrant des humains ; ces humains y sont à la fois spectateurs, acteurs et créateurs. Le projet SPI se décline selon 3 grands types d'application : les environnements virtuels de formation, l'analyse comportementale et la corésolution de problèmes entre agents et acteurs humains.
- **ARéVi** (Ateliers de Réalité Virtuelle) : Le projet ARéVi a pour objectif de définir et de développer des méthodes et des outils de prototypage interactif, collaboratif et coopératif, pour la mise en œuvre d'applications en environnements virtuels distribués. Les travaux que nous présentons dans cette thèse utilisent la plate-forme de réalité virtuelle de l'équipe ARéVi⁷.
- **EBV** (Ecosystémique et Biologie Virtuelles) : est l'équipe dans laquelle s'intègre ces travaux. Elle a pour objectif de modéliser et d'expérimenter *in virtuo* des phénomènes complexes de la biologie.

Cette brève description des domaines abordés confirme l'intérêt de la RV pour la cognition, la pédagogie et l'expérience. Ensuite, nous l'avons déjà cité, à l'instar du microscope, du télescope ou du macroscope de De Rosnay, le CERV nommé « virtuoscope » son outil conceptuel, qui repose sur une vision particulière de la RV. Le virtuoscope désigne alors un laboratoire virtuel pour l'étude des systèmes complexes, qui s'appuie sur les concepts, les modèles et les outils de la réalité virtuelle. La réalité virtuelle place l'utilisateur au cœur du laboratoire virtuel et le rapproche ainsi des méthodes des sciences expérimentales, tout en lui donnant accès aux méthodes numériques. La question qui se pose alors est : comment construire ce virtuoscope dans le cadre de l'équipe EBV et plus généralement dans celui de

⁶www.cerv.fr

⁷<http://sourceforge.net/projects/arevi/>

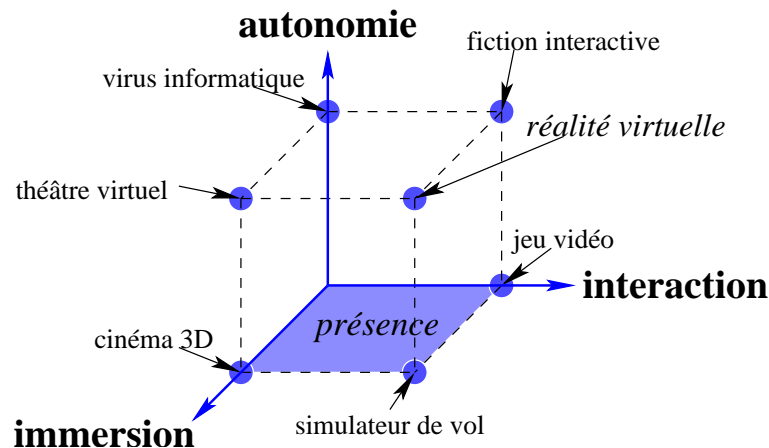


FIG. 3.3: Le principe d'autonomie s'ajoute aux deux dimensions : immersion et interaction pour définir un système de réalité virtuelle.

l'étude des systèmes biologiques ? En cela le principe d'autonomie constitue le premier élément de réponse.

3.3.2 L'autonomie

3.3.2.1 L'autonomie dans les univers virtuels

Le virtuoscope ajoute aux définitions classiques de la réalité virtuelle, la notion d'autonomie (voir figure 3.3) qui se présente à nous par essence, par nécessité, par ignorance et par conviction, pour construire nos mondes virtuels [Tisseau and Harrouet, 2003] :

- L'autonomie par *essence*, car nous cherchons à modéliser des systèmes composés d'entités, telles les cellules qui sont des systèmes autopoïétiques et qui sont donc, par définition, des systèmes autonomes [Varela, 1989a].
- Par *nécessité*, pour que les entités qui composent l'univers puissent s'adapter en cours d'exécution aux modifications des conditions aux limites (dues aux interactions, perturbations ou modifications imprévisibles de l'environnement).
- Par *ignorance*, car l'inexistence des modèles de comportement global de certains systèmes conduit à autonomiser les modèles des composants de ces systèmes. On souhaite pouvoir étudier des comportements émergents.
- Par *conviction*, en acceptant le partage du contrôle de l'évolution des univers virtuels entre les modèles numériques qui peuplent ces univers et les utilisateurs.

Ainsi le virtuoscope et par conséquent la manière de construire l'expérience *in virtuo* prennent pour principe l'autonomisation des modèles numériques et des entités qui peuplent les mondes virtuels. La méthodologie multi-agents semble alors tout à fait adaptée à la modélisation et à la simulation de phénomènes biologiques pour l'expérimentation *in virtuo*. En fait le principe d'autonomie permet de faire le lien entre l'ensemble des techniques classiques de la RV et les possibilités de mise en œuvre des systèmes dynamiques complexes offertes par les SMA. Pour cela, le modèle *Vowels* (présenté 2.4.5) peut être enrichi de la dernière voyelle, la lettre U comme Utilisateur [Tisseau, 2001]. Ce dernier étant modélisé, il peut alors interagir avec les autres agents en cours de simulation. Nous associons donc à la réalité virtuelle les propriétés déjà entrevues lors de la présentation des SMA (voir section

2.4.5).

En effet, si nous respectons ce principe d'autonomie lorsque nous définissons les modèles qui peuplent notre univers virtuel, nous pouvons concevoir de faire cohabiter ces modèles sans trop de difficultés. Leur autonomie leur permet de s'adapter et d'interagir avec l'environnement et les uns avec les autres, sans être dépendants les uns des autres. Nous obtenons ainsi des univers multi-modèles, ces modèles pouvant être de natures très différentes. En général, la modélisation des mécanismes biologiques diffère suivant l'angle sous lequel nous les observons. L'aspect multi-modèles (voir section 2.4.1) permet de faire cohabiter plusieurs points de vue pour décrire le vivant. Ainsi, pour modéliser des systèmes biologiques, il est possible de composer avec différents niveaux de modélisation, de granularité (voir section 2.3.5) ou avec différentes natures de modèle (chimique, géométrique, mécanique, etc...). De cette manière, on peut envisager une maquette numérique multi-modèles, multi-échelles et multi-métiers en combinant les points de vue de la biochimie, de la médecine, de la physiologie, de la biologie cellulaire ou moléculaire.

3.3.2.2 La médiation de l'esprit

La conception de la réalité virtuelle que nous décrivons permet d'entrevoir un niveau supplémentaire dans l'expérience. En effet, l'autonomie accordée aux modèles numériques doit permettre leur modification individuelle en cours d'exécution sans problèmes particuliers. Les lois et les phénomènes régissant l'univers virtuel étant modélisés, rien n'empêche alors de permettre à l'utilisateur de les modifier. Ainsi, au lieu d'agir sur le modèle en le perturbant (médiation de l'action), l'utilisateur peut modifier directement le modèle. Il passe alors du rôle d'acteur à celui de cré-acteur. On parle de la médiation de l'esprit (médiation de l'esprit de la cognition). Ce niveau d'expérience n'est en général accessible que *in abstracto* car l'expérimentateur ne peut changer les lois de la nature dans le réel. Lors d'une expérience, le scientifique teste la réactivité d'un système et peut mettre à jour la représentation qui énonce dans son esprit. Ici la réalité virtuelle permet de matérialiser en temps réel cette médiation de l'esprit, directement sur le modèle manipulé. Se faisant, elle offre un champ d'expérience plus vaste que dans la « réalité réelle ».

3.4 Le projet In Virtuo : Quels modèles ?

3.4.1 Réalité virtuelle et biologie

Un certain nombre de travaux concernant l'intérêt de l'utilisation de la RV pour l'étude des systèmes complexes du vivants ont déjà été réalisés au CERV. Cette section vise à les présenter. Il peut toutefois paraître un peu auto-centré de ne s'intéresser qu'à ces seuls travaux. Malheureusement, l'état de l'art de ce que nous appelons expérience *in virtuo* est assez anecdotique. En effet, seuls deux types de travaux relevant de l'appellation « réalité virtuelle et biologie » sont disponibles dans la littérature.

Tout d'abord, il existe des travaux liés à la médecine et parmi les plus aboutis ceux liés à la chirurgie. Citons par exemple l'environnement de RV pour la simulation chirurgicale du LIFL, à Lille [Meseure et al., 2003], et le projet suisse Co-ME⁸ qui rassemble un certain nombre de travaux d'aide à la chirurgie au moyen de systèmes de réalité augmentée⁹. On peut

⁸<http://co-me.ch/>

⁹Un système de réalité augmentée rend possible de superposer l'image d'un modèle virtuel 3D ou 2D sur une image de la réalité et ceci en temps réel.

également rattacher les techniques de téléchirurgie au domaine de la RV. Citons le simulateur chirurgical (*surgical simulator*) du *Rensselaer Polytechnic Institute*¹⁰. Ces applications sont malgré tout assez éloignées de notre sujet d'étude. Pourtant, l'idée de lier médecine et RV est assez « ancienne » puisqu'en 1998 l'article [Camp et al., 1998] présentait déjà un certain nombre de travaux dans le domaine, ils se limitent toutefois en général à la visualisation de données.

Il existe en effet, un certain nombre de travaux utilisant la seule possibilité d'immersion, limitant finalement la RV à un outil de visualisation 3D interactive. Ainsi l'Atlas¹¹ fournit des description visuelles des organes qui facilitent l'apprentissage de l'anatomie. La visualisation 3D interactive peut également se révéler intéressante pour appréhender la structure des vastes réseaux biochimiques [Yang et al., 2005] ou pour manipuler les immense banques de données biologiques [Rojdestvenski, 2003]. Les moyens techniques mis en œuvre pour l'immersion peuvent s'avérer assez impressionnants; ainsi certains projets utilisent un CAVE¹² [Karr and Brady, 2000].

Enfin, la notion d'expérience virtuelle en biologie est utilisée par le projet ViBE (*Virtual Biologie Experiment*) [Subramanian and Marsic, 2001] mais il s'agit d'un outil pédagogique ce qui encore une fois est assez loin de nos préoccupations.

Ainsi nous présentons, du point de vue de modélisation des entités biologiques, les sections suivantes les travaux réalisés au CERV. L'Habilitation à Diriger les Recherches de Vincent Rodin résume l'ensemble de ces projets [Rodin, 2004]. Nous les décrivons en trois étapes successives. Les premiers travaux définissent des agents entités (cellules et molécules). Ensuite la notion d'agent **Réaction** vient dépasser certaines limites du modèle microscopique. Enfin, nous montrons comment la notion d'organisation a été introduite dans les modèles.

3.4.2 Modèles microscopiques : agent Cellule et agent Molécule

Les travaux de Pascal Ballet [Ballet, 2000] ont cherché à exploiter l'analogie entre l'agent informatique et la cellule biologique. Ainsi, par similarité avec une cellule, un agent est doté d'un ensemble de récepteurs à sa surface. Un agent a également un comportement interne éventuellement assez complexe. Les agents sont soumis aux règles de l'environnement. Ces règles consistent à soumettre les agents, via leurs récepteurs, à l'influence des autres agents. Ces influences sont la cause de leurs mouvements de translation et de rotation. Lors des déplacements, des liaisons entre agents peuvent survenir. En fonction des stimuli reçus grâce à leurs récepteurs, les agents modifient leurs comportements et leurs états internes. Quand un agent **Cellule** est activé, il peut augmenter son nombre de récepteurs et inversement lorsqu'il est désactivé, un certain nombre de ses récepteurs sont éliminés par endocytose.

Les stimuli reçus grâce aux récepteurs correspondent, soit à la liaison avec les récepteurs d'autres agents **Cellule** soit, à la liaison avec des molécules. Lors de la modélisation d'une population de molécules, il est souvent impossible, pour des raisons de temps de simulation, d'associer une seule molécule à un agent. Afin de résoudre ce problème, un agent **molécule** représente un ensemble plus ou moins grand de molécules. De même, un agent **Cellule** peut être le représentant d'un certain nombre de cellules. Il est ainsi possible de passer de grandeurs quantitatives de simulation à des grandeurs quantitatives plus proches de la réalité.

Les liaisons entre agents peuvent aboutir à la formation de complexes. Un complexe, est lui-même un agent composé d'autres agents. Il est doté d'une masse et d'un centre de

¹⁰<http://www.rpi.edu/>

¹¹<http://www.bioanim.com/CellTissueHumanBody6/index.html>

¹²Un CAVE est un environnement d'immersion 3D pour un utilisateur. En général, c'est un cube dont certaines faces sont des écrans (parfois les six).

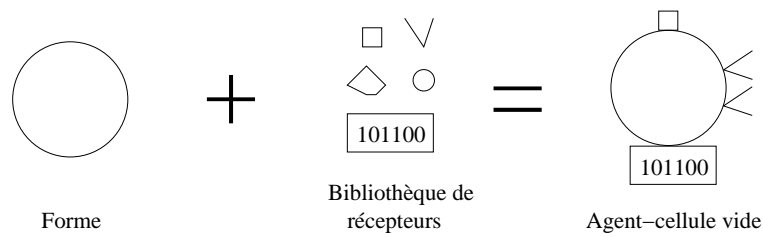


FIG. 3.4: Création d'un agent `Cellule` vide (d'après [Ballet, 2000]).

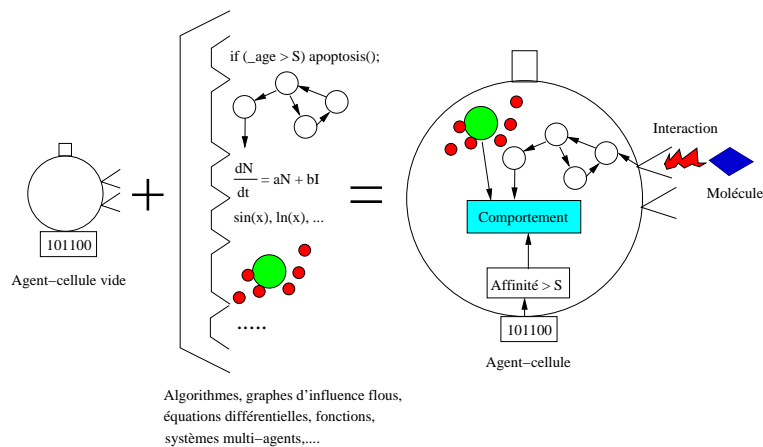


FIG. 3.5: Création d'un agent `Cellule` complet (d'après [Ballet, 2000]).

gravité. La masse du complexe est la somme des masses des agents qui le composent. Au sein du complexe, chaque agent conserve son indépendance comportementale, mais est soumis aux lois physiques imposées par le collectif. Cela signifie donc qu'un agent composé est une seule et même entité au vu des lois physiques de l'environnement.

3.4.2.1 Construction d'un agent `Cellule`

La construction d'un agent `Cellule` débute par le choix d'une forme représentant la membrane cellulaire. Celles retenues, les plus simples et les plus naturelles, sont le cercle en 2D et la sphère en 3D. D'autres formes sont également possibles mais plus complexes à gérer comme, par exemple, les ellipses ou les ovoïdes. Ensuite, des récepteurs choisis dans une bibliothèque sont placés autour de la forme-membrane (voir la figure 3.4). Ils peuvent être uniformément répartis à la surface de la cellule ou être dans des positions déterminées par leur position angulaire et leur distance par rapport au centre de l'agent. Cette disposition est importante dans la mesure où, en réalité, deux récepteurs proches à la surface d'une cellule peuvent interagir. Ainsi, les phénomènes d'auto-interaction entre récepteurs sont modélisables.

La troisième étape de la construction consiste à inclure les comportements internes. Ces comportements sont décrits par des algorithmes, des équations différentielles, des fonctions mathématiques, des systèmes multi-agents ou tout autre formalisme permettant de décrire des comportements (voir la figure 3.5).

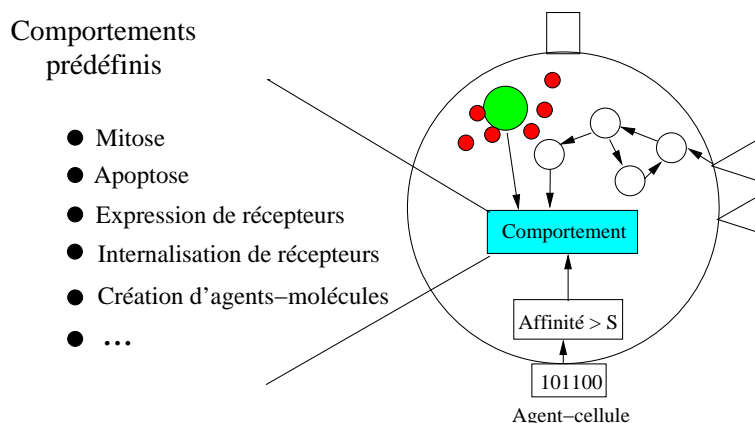


FIG. 3.6: Comportements de base prédéfinis (d'après [Ballet, 2000]).

3.4.2.2 Comportement d'un agent Cellule

Le choix entre algorithmes, graphes d'influence flous, équations différentielles, fonctions, systèmes multi-agents ou d'autres types de formalisme dépend de la nature de chaque problème au sein de la cellule, des connaissances acquises sur le problème ainsi que de l'habitude du modélisateur à manipuler l'un des outils plutôt qu'un autre.

Concernant les algorithmes, le modèle inclut un certain nombre de comportements prédéfinis. Par exemple, un agent a la possibilité de produire des récepteurs à sa surface (pour devenir sensible à un type de molécule), d'internaliser ses récepteurs lors de liaisons avec des molécules, de se diviser (mitose), de créer de nouveaux agents *Molécule* (agent *Interleukine*, *Interféron*,...) ou de se détruire (apoptose). D'autres comportements spécifiques sont intégrables afin d'améliorer et d'affiner le modèle (voir la figure 3.6).

Rappelons que pendant la simulation, des agents peuvent se regrouper, formant alors un agent composé. Un tel agent est considéré par l'environnement comme une seule et même entité pour ses déplacements, mais reste une mosaïque de comportements individuels. Un agent composé peut se fractionner ou fusionner avec un autre agent composé. Cela survient lors de création ou de destruction de liaisons.

3.4.2.3 Interactions entre agents Cellule, agents Molécule

Les interactions entre agents se font donc à travers les récepteurs des agents. Les liaisons entre récepteurs ne sont pas définitives, elles peuvent se défaire dans trois cas. Premièrement, lorsqu'une des cellules d'un complexe meurt, les différents récepteurs sont libérés et sont donc à nouveau capables de se lier à d'autres récepteurs. Le second cas survient lorsque la durée de la liaison atteint une valeur déterminée. En effet, les liaisons moléculaires sont temporaires : leur durée est mesurée en demies-durées de vie. C'est-à-dire que pour un ensemble de liaisons de même type, au bout de la demie-durée de vie, la moitié des liaisons intermoléculaires est détruite. Le dernier cas correspond à la liaison entre un agent *Cellule* et un agent *molécule*. L'agent *Cellule* internalise alors l'entité représentant la molécule. L'internalisation élimine la molécule au sein de la cellule après que celle-ci ait émis un signal.

Les forces d'attraction entre les agents prennent en compte la taille et la masse des objets (cellules, molécules) ainsi que l'affinité entre les récepteurs [Oprea and Forrest, 1998]. Notons que dans le cas de complexes, les forces totales d'attraction doivent être calculées. Ainsi, à partir de ces forces d'attraction, les agents se déplacent alors en translation et en rotation.

3.4.2.4 Intérêt et utilisation des agents Cellule

L'utilisation des agents **Cellule** nous semble appropriée dans essentiellement deux cas. Tout d'abord, lorsqu'on cherche à modéliser un phénomène où les cellules impliquées sont relativement bien connues dans leurs comportements de base et dans leurs interactions. Ensuite, comme nous venons de le voir dans l'exemple précédent, cette approche peut être intéressante lorsqu'on cherche à modéliser l'intérieur d'une cellule et que l'on veut comprendre l'impact d'une modélisation intra-cellulaire sur un ensemble de cellules.

Cette approche a été utilisée dans plusieurs domaines. Citons, notamment, les domaines de l'immunologie avec l'étude de l'apoptose, de l'hématologie avec la modélisation de la coagulation du sang ou de la cancérologie avec la modélisation de voies d'activation de la cellule. Il se trouve néanmoins que quelques limites en terme de puissances de calcul se sont posées avec cette méthode. En effet les niveaux de granularité choisis pour représenter les molécules ont conduit à développer l'idée d'agents **Réaction**.

3.4.3 Agents Réaction et Diffusion

3.4.3.1 Historique

Le second volet des travaux concerne le principe des agents **Réaction** introduit lors de la thèse de Sébastien Kerdélo [Kerdélo, 2006]. Comme nous venons de l'indiquer, ce principe a été développé pour pallier une lacune du modèle que nous venons de présenter. En effet les molécules y sont modélisées par des agents, au même titre que les cellules. Le problème est que la quantité de molécules qui sont mises en jeu est considérable par rapport au nombre de molécules ; à tel point qu'il est impossible d'attribuer un agent à chaque molécule présente.

La première solution qui est apparue a été de considérer qu'un agent **Molécule** représente un certain paquet de molécules. Ainsi le nombre d'agents peut devenir presque acceptable. Mais cette simplification pose divers problèmes pour passer du qualitatif au quantitatif.

La solution classique devrait alors être un changement d'échelle : passer du niveau microscopique au niveau macroscopique. Dans ce cas il faut modéliser le comportement de la population plutôt que celui de l'individu. Il y a alors un agent **Population**, dont le comportement est régi par une équation différentielle. Le problème est que, soit nous connaissons l'évolution *a priori* des populations de molécules, soit cette équation dépend de l'ensemble des autres agents du milieu agissant avec la population. Ce dernier choix implique qu'à chaque modification du SMA, il faut recalculer toutes les équations des comportements de chaque population, ce qui porte un coup fatal au principe d'autonomie. De plus cette façon de faire est strictement équivalente à une résolution classique d'équation différentielle, qui va à l'encontre de l'esprit individus-centré de la simulation par SMA.

Si la modélisation des molécules n'est applicable ni au niveau microscopique, ni au niveau macroscopique, alors peut-être faut-il changer l'objet de la modélisation ? C'est l'idée des agents **Réaction** qui visent à modéliser le phénomène de la réaction chimique (plutôt que les molécules) par un agent.

3.4.3.2 Principe de fonctionnement

L'agent autonome **Réaction** réalise le cycle à trois temps, perception, décision, action :

- Lecture des concentrations des espèces chimiques impliquées dans le phénomène ;
- Calcul de la vitesse de réaction et intégration sur le pas de temps ;

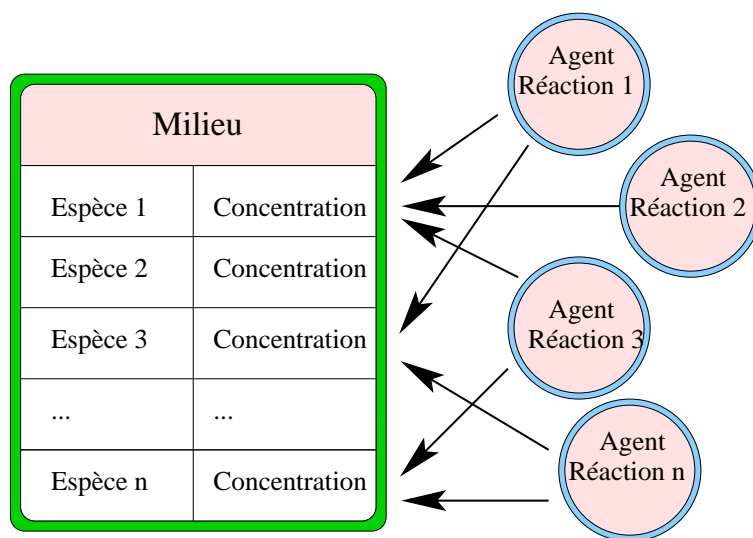


FIG. 3.7: Principe des agents **Réaction** (d'après [Kerdélo, 2006]) : à tour de rôle, chaque agent perçoit et modifie les concentrations des espèces chimiques pour faire évoluer le système.

- Mise à jour des concentrations des réactifs et des produits de la réaction.

La figure 3.7 montre le principe de ces agents. Un système composé d'agents **Réaction** peut ainsi modéliser et simuler n'importe quel système de réactions chimiques. Ils ont notamment servi à simuler des cascades de réactions chimiques impliquées dans les mécanismes de la coagulation du sang en mettant en œuvre 42 réactions et 34 espèces. Ils ont alors montré qu'il était possible de s'attaquer à des problèmes raides.

Évidemment, cette méthode reste moins efficace qu'une résolution classique d'équation différentielle, mais présente l'intérêt d'être adaptée à la vision SMA et de respecter le principe d'autonomie.

3.4.3.3 Formalisation et validation

Considérons un milieu homogène sans dimension, contenant quelques réactifs. Soit $[\vec{C}(t)]$ un vecteur de concentrations à l'instant t . Dans ce milieu, m réactions interviennent. Leurs vitesses respectives sont données par les fonctions vectorielles f_i , $1 \leq i \leq m$, dont les arguments sont le temps et le vecteur des concentrations. L'évolution des concentrations de réactifs dans le temps est classiquement décrit par le système différentiel :

$$\frac{d}{dt}[\vec{C}(t)] = (f_1 + f_2 + \dots + f_m)(t, \vec{C}[t]),$$

sous les conditions $\vec{C}[t_0]$, les concentrations à l'instant initial. Ce type de système peut être résolu précisément par les méthodes présentées section 2.1.1, qui permettent le calcul des concentrations à tous les instants du temps discrétisé : pour les méthodes à un pas, le vecteur de concentration \vec{C}_{n+1} à l'instant t_{n+1} est calculé à partir du même vecteur nommé \vec{C}_n à l'instant t_n . Cela conduit à l'exécution d'un algorithme comme suit :

$$\begin{aligned} \vec{C}_0 &= \vec{C}[t_0] \\ \vec{C}_{n+1} &= \vec{C}_n + h_n \Phi_{f_1+\dots+f_m}(t_n, \vec{C}_n, h_n) \end{aligned}$$

où $h_n = t_{n+1} - t_n$, $\Phi_{f_1+\dots+f_m}$ est une fonction dépendante de la somme des vitesses f_i qui caractérise l'algorithme choisi. Comme nous venons de le signaler, toutes les réactions chimiques s'exécutent simultanément. Nous l'avons déjà indiqué, l'inconvénient principal de cette modélisation est qu'elle est statique : l'ajout ou la suppression d'une réaction en cours d'exécution conduit à la réécriture du système et à la réexécution du programme. La méthode des agents **Réaction** utilise aussi les algorithmes classiques de résolution, mais les applique pour chaque réaction durant un même pas de temps : ainsi, à chaque algorithme classique correspond une version agent **Réaction** (pour une formalisation précise, voir [Redou et al., 2005]). Nous présentons ici un exemple simple, ainsi que les principaux résultats à propos de la convergence. Lors d'un unique pas de temps, l'algorithme est appliqué m fois : une fois par réaction. Chaque application prend en compte l'état du système au temps courant. Pour éviter l'apparition d'un biais, un ordonnancement aléatoire des exécutions des agents est réalisé à chaque pas de simulation.

Les résultats principaux à propos de la convergence de la méthode des agents **Réaction** sont les suivants :

- La convergence de la version agent **Réaction** de la méthode d'Euler est en moyenne d'ordre 1.
- La convergence de la version agent **Réaction** de la méthode de Runge-Kutta d'ordre 2 est en moyenne d'ordre 2.
- Considérant une méthode à un pas d'ordre de convergence ≥ 3 , la version agent **Réaction** de la méthode est convergente d'ordre 2.

Ajoutons que la méthode a également été adaptée à des méthodes à pas multiples et à des méthodes implicites. Les propriétés de convergence et de stabilité sont alors meilleures que pour les méthodes à un pas. Cependant elles sont en conflit avec la possibilité d'interagir avec le modèle à tout moment de l'exécution.

3.4.3.4 La diffusion

Avec le même principe, il est possible d'introduire des dimensions spatiales et de considérer la diffusion moléculaire entre milieux homogènes en trois dimensions (voir figure 3.8). Rappelons que ce genre de problème est classiquement résolu au moyen d'équations aux dérivées partielles :

$$\frac{\partial C}{\partial t}(x, y, z, t) = D \left(\frac{\partial^2 C}{\partial x^2} + \frac{\partial^2 C}{\partial y^2} + \frac{\partial^2 C}{\partial z^2} \right) (x, y, z, t) \quad (3.1)$$

où $C(x, y, z, t)$ est la concentration de réactif en (x, y, z) , au temps t , et D le coefficient de diffusion. Si nous souhaitons appliquer la médiation de l'action et conserver le principe d'autonomie, les équations aux dérivées partielles sont trop rigides. En conséquence, la méthode des agents **Diffusion** a été introduite comme suit.

L'espace en 3 dimensions est discrétisé. Pour simplifier, nous considérons deux mailles de même taille, entre lesquelles un agent **Diffusion** réalise un transfert d'énergie ajoutant d'un côté la quantité de réactif qu'il enlève de l'autre, tout en prenant en compte le coefficient de diffusion du réactif. Si le réactif diffuse de gauche à droite, l'action de l'agent **Réaction** peut être décrite par les équations :

$$C_d(t + \Delta t) = C_d(t) - D(C_d(t) - C_g(t)) \cdot \Delta t \quad (3.2)$$

$$C_g(t + \Delta t) = C_g(t) + D(C_d(t) - C_g(t)) \cdot \Delta t \quad (3.3)$$

avec C_g (respectivement C_d) la concentration à gauche (respectivement à droite) et D le coefficient de diffusion. Lorsque l'espace est discrétisé, il y a autant d'agents **Diffusion**,

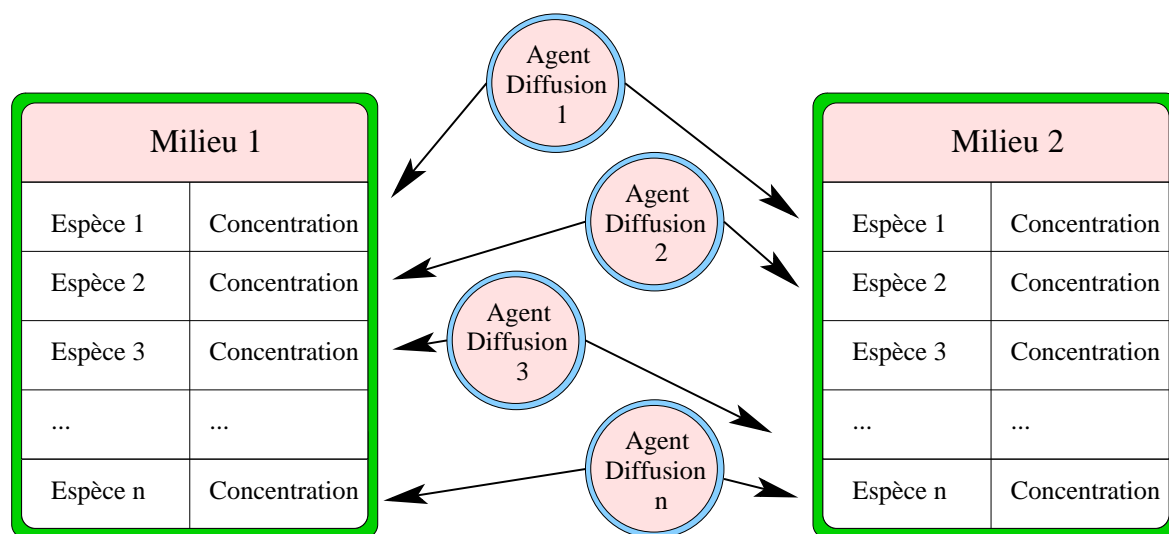


FIG. 3.8: Principe des agents *Diffusion* : à tour de rôle, chaque agent perçoit et transporte les concentrations des espèces chimiques d'un milieu à l'autre.

qu'il y a de faces communes aux mailles composant le milieu. De la même manière et pour les mêmes raisons que pour les agents *Réaction*, les agents *Diffusion* agissent au moyen d'itérations chaotiques et asynchrones (voir section 4.2). Les résultats suivants ont été démontrés : comme la méthode des différences finies pour résoudre l'équation (3.1), la méthode des agents *Diffusion* basée sur les équations (3.2 et 3.3) converge à l'ordre 2 dans l'espace et à l'ordre 1 dans le temps.

Finalement nous possédons deux outils efficaces pour décrire les phénomènes chimiques au sein des modèles pour l'expérience *in virtuo* : les agents *Réaction* et *Diffusion*. Ces outils peuvent alors être intégrés dans des modèles d'étude de phénomènes biologiques complexes.

3.4.4 Agents Organisation

La thèse de Gabriel Querrec [Querrec, 2005] a porté sur l'étude d'une maladie complexe : le myélome multiple par l'utilisation de l'outil *in virtuo*. Le myélome est un cancer qui se traduit par une prolifération tumorale des lymphocytes dans la moëlle osseuse. La modélisation et la simulation de deux niveaux de description combinés, permet d'avancer dans la compréhension du comportement de la tumeur cancéreuse. Ces deux niveaux sont : une modélisation des lymphocytes, des ostéoclastes et ostéoblastes par des agents *Cellule* (voir figure 3.9), et une modélisation des voies de transduction (communication intra-cellulaire) et des voies extracellulaires par des agents *Réaction* (voir figure 3.10). Ces travaux ont permis de dégager de nouvelles hypothèses concernant les mécanismes de (dé-)régulation des lymphocytes et donc des tumeurs.

Ces travaux ont proposé un premier modèle basé sur une association ternaire des classes *Interaction*, *Entité* et *Organisation*. Les agents *Cellule* dérivent ainsi de la classe *Entité* et les agents *Réaction* de la classe *Interaction* (voir figure 3.11).

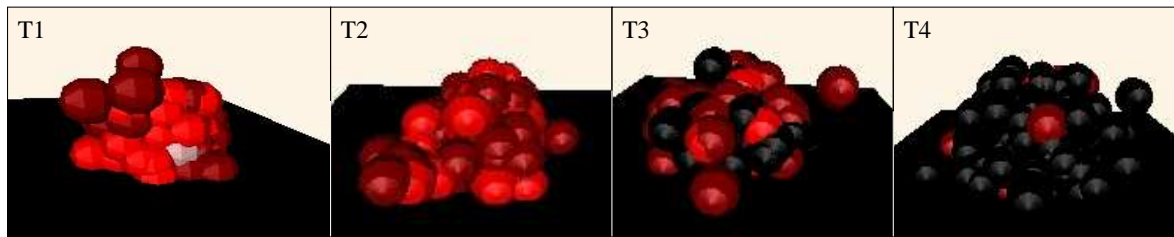


FIG. 3.9: Capture d'écran de la simulation d'une tumeur myélomateuse (d'après [Querrec, 2005]).

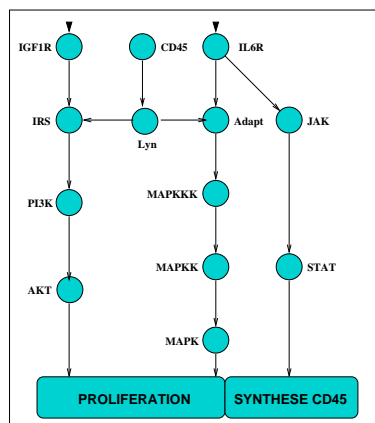


FIG. 3.10: Voies de signalisation intra-cellulaires des cellules myélomateuses (d'après [Querrec, 2005]).

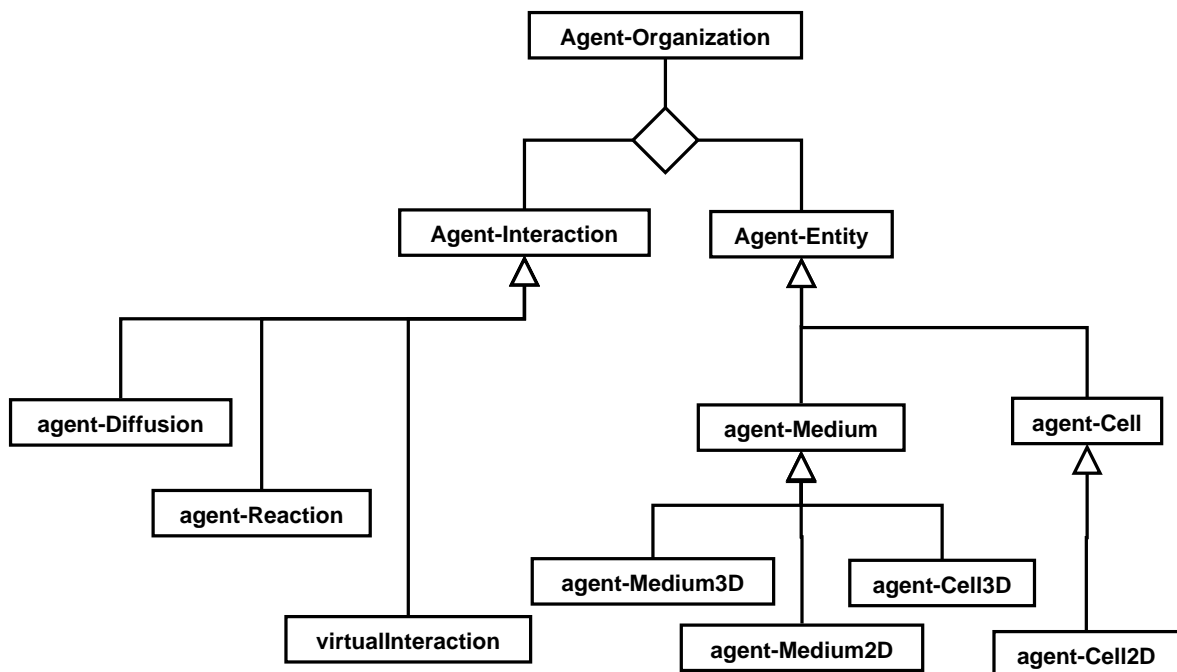


FIG. 3.11: Diagramme de classes introduisant la notion d'organisation.

3.5 Le besoin d'une architecture générique

Dans cet état de l'art, nous avons d'abord voulu donner quelques éléments pour comprendre les attentes de la recherche en biologie. Nous avons constaté que cette science expérimentale possède à l'origine assez peu d'outils théoriques, les rares présents étant empruntés aux autres domaines scientifiques. Ensuite, nous avons décrit l'émergence de la biologie moléculaire et l'explosion des données due à la génétique. C'est à ce moment que l'informatique est entrée en scène avec la bio-informatique. Enfin nous avons fait état de l'émergence d'un nouveau paradigme en biologie que nous appelons biologie théorique, systémique ou intégrative. Il paraît essentiel à l'informaticien souhaitant œuvrer pour la modélisation des systèmes vivants de prendre en compte l'histoire des différents paradigmes de la biologie. Ceci est d'autant plus vrai que la demande en outils de modélisation et de simulation provient du changement profond dans la manière dont sont conçus les systèmes complexes vivants par les biologistes.

La deuxième partie de cet état de l'art a consisté à présenter un tour d'horizon des différentes techniques de formalisation et de simulation mises à dispositions de la biologie par le domaine de l'informatique. Parmi les formalismes, les algèbres de processus, les réseaux de Petri et la logique temporelle permettent de manipuler les réseaux complexes de la biologie. Ensuite, nous nous sommes intéressés plus particulièrement aux outils de simulations, proposant un nouveau moyen d'investigation : les calculs *in silico*. Plusieurs méthodes mettent en œuvre ces calculs. Nous avons préféré les méthodes orientées objets et agents car elles permettent la simulation multi-échelles et multi-modèles plus facilement que les outils classiques reposant sur la résolution d'équations différentielles.

Le dernier chapitre a eu pour objectif de montrer qu'il est possible d'aller plus loin dans l'acte d'expérience revendiqué par les concepteurs d'expérimentation *in silico*. Le principe de l'énaction nous a fourni quelques concepts théoriques pour comprendre et définir la cognition. Les concepts de la réalité virtuelle et le lien fort que ce domaine entretient avec l'acte d'expérience nous permettent alors d'envisager un nouveau type d'expérimentation des modèles numériques, l'expérience *in virtuo*. Cette dernière passe nécessairement par la double médiation des sens et de l'action. Nous avons ensuite montré que le principe d'autonomie se révèle incontournable pour construire les univers virtuels. Se faisant, la réalité virtuelle offre également un troisième type de médiation : la médiation de l'esprit.

Les différents travaux réalisés au CERV tendent à mettre en pratique les concepts et les outils de la réalité virtuelle pour modéliser et expérimenter les systèmes complexes vivants. Ainsi nous avons présenté les agents **Cellule**, **Réaction** et **Organisation**.

L'idée de mélanger les agents **Cellule** et les agents **Réaction**, introduite par les travaux sur le myélome multiple, a permis d'avancer dans l'établissement d'expériences *in virtuo* de systèmes multi-modèles. Cependant, il apparaît que ces différents modèles reposent sur deux visions de la modélisation peu compatibles dans le cas général et surtout redondantes. En effet, pour les agents **Cellule**, le modélisateur s'appuie sur la notion d'entité, alors que pour les agents **Réaction**, ce sont les phénomènes qui sont modélisés sous la forme d'interactions. L'algorithme assurant la dynamique d'un phénomène particulier est alors réalisé par l'entité active ou par l'interaction active. Il ne peut en aucun cas être réalisé par les deux sous peine de dupliquer la dynamique du système. Le modélisateur doit donc décider au cas par cas, quelle solution il souhaite appliquer.

La souplesse du paradigme agent permet de mélanger ces deux visions. Mais, le manque de cadre strict peut très rapidement conduire à l'écriture de code non maintenable. De plus, le manque de généricité freine la modularité dans les applications. Ces difficultés limitent

l'évolution de nos applications en rendant difficile la réutilisation de code d'un modélisateur à l'autre. Un cadre générique plus strict permettrait de pallier ces défauts.

Le travail que présente cette thèse intervient donc dans la continuité des travaux du CERV. Il vise à définir un cadre générique permettant de rassembler les différentes techniques déjà développées, au sein d'une même boîte à outils. Pour cela, nous devons faire un choix dans la posture de modélisation. C'est la vision « interaction » que nous choisirons. La seconde partie de la thèse expose donc le modèle générique proposé. Elle se divise en deux chapitres. Tout d'abord, nous définissons le cœur de ce que propose cette thèse. Pour cela, le chapitre 4 est donc consacré à la définition du modèle générique. Ensuite, nous montrons comment le modèle peut être dérivé et appliqué aux différents phénomènes biologiques qui nous intéressent. C'est le rôle du chapitre 5 qui présente une « boîte à outils » biologique pour l'expérience *in virtuo*.

Deuxième partie

Modèles

Chapitre 4

Le modèle générique

Comment construire nos applications pour l'expérimentation *in virtuo*? Voilà finalement la question posée par notre état de l'art. Le but de cette thèse n'est donc pas de définir la nouvelle théorie biologique ou le nouveau formalisme du vivant, mais de proposer une méthode générique de modélisation fournissant un bon compromis compatible avec :

- les réalités de la biologie appliquée,
- les évolutions théoriques proposées par la biologie intégrative ou systémique,
- notre conception de la réalité virtuelle,
- le patrimoine du projet *in virtuo* du CERV.

Pour se faire nous proposons ici un modèle générique. L'utilisation de la programmation orientée objet convient au principe de la modularité que nous nous imposons. Le contexte de l'équipe EBV et plus généralement du CERV nous incite également à utiliser une programmation orienté agent. Nous le verrons, l'utilisation du langage C++ nous permet d'utiliser la bibliothèque ARéVi et de concevoir des agents sous la forme d'objets actifs gérés par un ordonnanceur.

Ce chapitre est alors construit de la manière suivante. Premièrement, nous fondons notre modèle sur la réification des interactions en objets actifs autonomes pour permettre une conception multi-modèles. Ensuite, nous exposons les différentes solutions d'ordonnement asynchrone des objets autonomes pour permettre la simulation multi-échelles temporelles. Enfin, nous intégrons les notions d'organisation et de phénomène pour permettre une construction modulaire de systèmes autonomes.

4.1 Réification des interactions

4.1.1 Motivations

Nous traitons ici de ce qui constitue l'originalité et le point de départ de cette thèse. Ce travail repose en effet sur une idée assez simple qui consiste à changer la posture classique de modélisation centrée sur les entités pour en adopter une autre basée sur la réification des interactions entre ces entités. Plusieurs raisons motivent ce choix.

Premièrement, lors des travaux sur les agents cellules et molécules, la modélisation microscopique des composants a montré ses limites en terme de puissance de calcul. Pour dépasser ce problème, les agents `Réaction` modélisent les réactions chimiques plutôt que les composants

Molécule (voir section 3.4.3). Nous parlons alors de « réification » des réactions chimiques. L'acception classique du terme « réification » en programmation orientée objet, signifie que l'on définit une classe correspondant à ce que l'on réifie. Réifier les réactions chimiques signifie donc qu'on les « objectifie » : on crée la classe **RéactionChimique**. Un des avantages de cette méthode est qu'elle permet de prendre en compte un niveau de modélisation plus macroscopique tout en conservant l'intérêt d'une architecture individu-centrée et orientée agent.

Il est également plus élégant, au niveau de l'implémentation, de réifier l'interaction si elle peut l'être. En effet, cela permet d'éviter la redondance de la description et de l'application des protocoles d'interaction¹. De plus, cela permet la réalisation d'une dynamique multi-échelles temporelle (ce sera l'objet de la section suivante).

Ensuite, l'utilisation de l'approche biologie systémique, intégrative ou théorique (voir section 1.5) pour l'étude des systèmes complexes, nous pousse à nous focaliser sur les relations entre les composants. D'ailleurs, cette importance donnée aux relations est un point commun à l'ensemble des théories étudiant les systèmes complexes ; ce qui nous donne une raison supplémentaire de rendre explicite ces interactions pour leur donner au sein de nos modèles numériques, la même importance que dans les modèles théoriques.

Enfin, malgré l'intérêt *a priori* de cette posture, l'examen de notre état de l'art indique qu'à l'exception des travaux sur les agents **Reaction** [Kerdélo, 2006], cette voie ne semble pas avoir été explorée. La majorité des applications sont en effet centrées composant (ou entité) et aucune n'est centrée interaction. Ainsi, nous choisissons de construire nos modèles autour de cette idée de réification des interactions, la notion d'objet **Réaction** se généralisant en objet **Interaction**.

Interrogeons-nous alors logiquement, sur l'intégration de ce choix dans nos univers autonomes.

4.1.2 Abstraction des composants

Nous avons défini la réification comme l'« objectification ». Mais, nous nous plaçons ici dans le contexte d'applications prenant pour principe la notion d'autonomie. Autrement dit pour aller au bout de cette démarche, il nous faut conférer une certaine autonomie aux objets **Interaction**. Enrichissons alors le sens du verbe réifier pour l'adapter à notre contexte de réalité virtuelle : réifier signifie dorénavant « objectifier et autonomiser ». Nous obtenons ainsi des agents **Interaction**.

Que faisons nous maintenant de nos agents **Composant** (modélisant les cellules, les molécules...) ? La réification des interactions entraîne la délocalisation de ce qui fait la dynamique du système, des objets **Composant** vers les objets **Interaction**. De ce fait, l'activité des agents **Composant** devient nulle. Nous observons une abstraction des composants qui se réduisent alors à de simples états de la mémoire servant de médiateur aux interactions. La figure 4.1 illustre ce changement de point de vue.

¹Lorsque deux entités, A et B, doivent être en relation, nous ne savons si c'est A qui agira sur B, ou si c'est B qui agira sur A. Dans tout les cas, le protocole d'interaction doit être décrit concrètement quelque part :

- S'il est décrit dans A et dans B, il y a redondance de la description et potentiellement de la mise en œuvre de l'interaction, ce qui n'est bien sûr pas souhaitable.
- S'il n'est décrit que pour l'une des deux entités, nous parlons alors plus d'action que d'interaction. Ceci pose un problème, car savoir si A agit sur B ou B agit sur A est souvent indécidable.
- S'il est décrit dans une troisième entité C, alors le problème de redondance ne se pose plus. L'entité C représente finalement l'interaction réifiée.

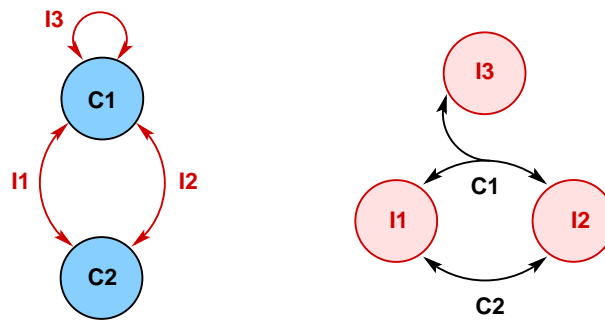


FIG. 4.1: Ces deux graphes duaux illustrent le changement de point de vue opéré grâce à la réification des interactions. À gauche, le graphe représente la méthode classique de modélisation d'un système avec des nœuds "composants" et des arcs "interactions". Le graphe de droite représente notre approche dans laquelle les arcs sont transformés en nœuds et inversement.

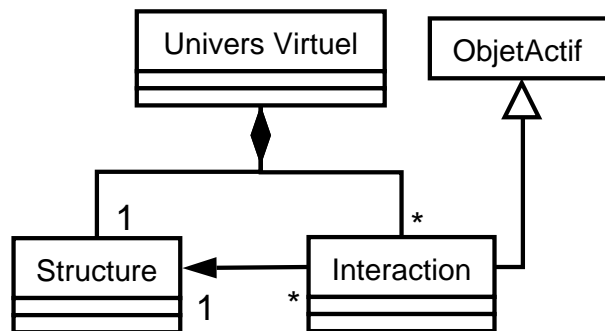


FIG. 4.2: La classe `Univers Virtuel` est composé d'un état du monde (classe `Structure`), et d'objets actifs réalisant sa dynamique (classe `Interaction`).

4.1.3 Point de départ du modèle informatique

L'ensemble des composants constitue la structure du système modélisé dans l'univers virtuel. L'ensemble des objets actifs `Interaction` représente ce qui fait la dynamique du système. Nous reviendrons sur cet ensemble dans la section 4.3. Le diagramme de classe de la figure 4.2 correspond à notre approche. Le modèle est alors une sorte d'architecture en tableau noir. En effet, une instance de la classe `Structure` est un objet passif qui n'est qu'un ensemble d'états du monde et qui, par conséquent, peut être considéré comme un tableau noir de mémoire partagée par les instances de la classe `Interaction`.

Dans un souci d'autonomie, il paraît souhaitable que les objets actifs `Interaction` ne connaissent que le sous-ensemble de la structure qui les concerne. La classe `Structure` est alors un conteneur se composant de constituants, chaque objet `Constituant` représentant un morceau de l'état du monde. Une interaction connaît seulement l'ensemble des constituants qui la concerne. Le diagramme de la figure 4.3 montre l'intégration de la nouvelle classe `Constituant`.

Voici donc poser la première pierre de notre modèle. Il est maintenant intéressant de nous interroger sur notre positionnement informatique.

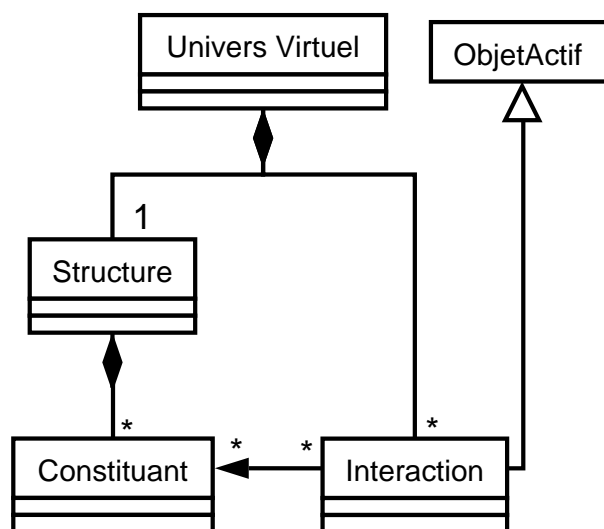


FIG. 4.3: La classe `Univers Virtuel` est composée d'un état du monde (classe `Structure`), et d'objets actifs réalisant sa dynamique (classe `Interaction`). Une `Structure` est composée de `Constituants`. Chaque interaction agit sur un certain nombre de constituants.

4.1.4 Agents ou objets actifs ?

Le modèle basé sur la réification des interactions ressemble à une architecture en tableau noir. Durant l'état de l'art, nous avons considéré ce type d'architecture comme faisant partie de la famille des agents réactifs². Nous nous interrogeons ici, sur l'aspect « système multi-agents » du modèle.

Tout d'abord, si il y a un candidat à l'appellation agent, c'est bien l'objet de type `Interaction`. Il n'est soumis à aucun contrôle global, il est autonome. C'est un objet actif qui a des activités de :

- perception : l'état des constituants,
- décision : calcul de la dynamique à mettre en œuvre,
- action : modification de l'état des constituants.

Ensuite, l'environnement de l'interaction est à la fois composée de la structure et des autres interactions. Mais, l'interaction ne peut en aucun cas percevoir l'ensemble de cet environnement. Elle ne perçoit que le sous-ensemble immuable de constituants qui lui est imparti.

L'interaction entre les agents est une notion clé des SMA. Nous parlons ici d'« interactions entre les `Interactions` », ce qui ne veut plus dire grand chose. Les liens entre les `Interactions` sont réalisés grâce aux constituants, donc indirectement par l'intermédiaire de l'environnement. En fait les objets `Interaction` communiquent indirectement par couplage des ensembles structuraux (les constituants) auxquels ils sont associés. Nous n'utilisons donc pas non plus, la communication par envoi de messages.

Enfin, les systèmes multi-agents sont l'apanage des simulations individu-centrées. Or, nous ne sommes plus tout à fait dans cette démarche de modélisation, même si d'un point de vue informatique, l'architecture peut correspondre à ce type d'approche.

²Ce classement peut d'ailleurs être mis en cause étant donné que certains travaux sur l'architecture en *blackboard* ne mentionne pas l'appellation agent. De plus, l'architecture en *blackboard* a précédé la notion d'agent.

Ces constatations nous amènent aux limites de la définition *Vowel* présentée à la section 2.4.5. La notion d'agent cognitif serait, elle, totalement inadaptée. Celle d'agent réactif peut correspondre à la limite, mais elle ne nous apporte pas grand chose étant donnée la distance avec les travaux classiques. De plus, elle véhicule un certain nombre de préconceptions et de suppositions qui sont aussi variées qu'il y a de concepteurs de systèmes multi-agents. Finalement, nous préférons abandonner la dénomination agent pour utiliser l'appellation « objet actif » ou « entité autonome ».

Les bases de notre modèle sont ainsi posées : nous souhaitons modéliser les systèmes vivants complexes au moyen d'entités autonomes réifiant les interactions façonnant l'état des systèmes. Il nous faut maintenant décrire le mode d'exécution de l'activité des objets *Interaction*.

4.2 L'ordonnancement asynchrone multi-échelles

4.2.1 L'ordonnancement

À chaque objet actif correspond un processus. Dans les articles présentant les applications orientées agents pour la biologie, la question de l'ordonnancement de ces processus est le plus souvent éludée. Il est implicitement considéré que le modèle numérique est strictement équivalent au modèle théorique. L'ordonnancement est alors perçu comme un détail d'implémentation qu'il n'est pas nécessaire d'explicitier. Or, nous le verrons, le choix de la méthode d'ordonnancement constitue une hypothèse forte de la modélisation. Il pose en fait la question du temps, qui n'est pas des moindres dans le contexte de la modélisation et de l'étude de systèmes dynamiques. Trois points doivent alors être abordés : la notion de cycle de simulation, le temps réel et la synchronicité.

4.2.1.1 La notion de cycle

L'ordonnanceur est un programme qui a la tâche de faire évoluer les objets actifs. Nous considérons que sa mission revient à appeler successivement une fonction que l'on appelle l'activité, celle-ci étant une fonction particulière associée à chaque objet actif. C'est elle qui est en charge de faire *vivre* cet objet sur le pas de temps qui lui est imparti.

Un cycle de simulation correspond au passage du temps virtuel, de l'instant t , à l'instant $t + 1$. Il correspond donc à la réalisation d'un pas de temps pour la simulation. Dans le cas simple, un cycle correspond à l'appel unique de chaque activité du système.

Nous commençons notre réflexion en choisissant d'utiliser cette notion de cycle par défaut car elle très majoritairement employée. Nous constaterons malgré tout qu'elle n'est pas indispensable.

4.2.1.2 Temps réel et temps virtuel

Deux temps doivent être distingués pour éclaircir le discours. D'abord, le temps réel ou temps physique correspond à celui que nous subissons, celui de notre montre. Ensuite le temps simulé ou temps virtuel correspond au temps de la simulation informatique, celui que « perçoivent » les entités autonomes *Interaction* et qui s'écoule dans l'univers virtuel. Donner une définition plus précise de ces deux temps demanderait de définir la notion de temps.

Ce n'est pas l'objet de la thèse. Au long du chapitre, nous formulerons quelques remarques pour enrichir la notion de temps virtuel. Ceci étant dit, il nous faut présenter deux modes d'exécution des simulations.

Le mode d'exécution temps réel³ signifie que l'on maîtrise la durée d'exécution des programmes. Ainsi il devient possible de faire correspondre le temps réel au temps simulé. Évidemment cette correspondance sera toujours imparfaite puisqu'il s'agit ici de faire correspondre un temps discret (virtuel) avec un temps continu (réel). Donc plus la période d'un cycle sera faible, meilleure sera la correspondance.

Pendant le mode d'exécution temps réel, le temps simulé est fonction du temps réel. Cette fonction peut être l'identité (à une latence près), mais elle peut également être toute autre. Ainsi pour permettre l'expérience d'un phénomène trop rapide dans le cas réel, il est possible de décréter qu'une seconde de temps virtuel correspond à 100 secondes de temps réel. À l'inverse, pour expérimenter une dynamique extrêmement lente il est possible d'accélérer. Dans tout les cas, pour prétendre pratiquer des expériences de réalité virtuelle, nous aspirons à respecter la contrainte du temps réel. Cela signifie également que la latence entre la réalisation d'une action par l'utilisateur et sa prise en compte dans le monde virtuel par les modèles autonomes qui le compose soit « acceptable ».

La quantité de calculs à réaliser par unité de temps virtuel varie au cours de l'évolution du système simulé. L'application de la contrainte temps réel implique que cette quantité doit être bornée pour assurer le lien entre le temps virtuel et réel. Dans ce cas, l'application doit nécessairement se trouver dans l'une ou l'autre des situations suivantes :

- La quantité de calculs est inférieure à la limite. L'application se met en attente (oisive ou *idle*) pour que le temps réel « rattrape » le temps virtuel.
- La quantité de calculs est supérieure à la limite. L'application remet à plus tard certains appels d'activité pour que le temps virtuel « rattrape » le temps réel.

La première situation peut être considérée comme une perte de temps. La seconde introduit une perte de précision au niveau de la simulation. Parfois, c'est plus l'aspect expérimentation que celui de l'expérience qui peut être souhaité. Donc, dans un souci de concilier la validité des calculs et l'optimisation du temps de calcul, l'utilisateur peut être amené à utiliser le mode temps virtuel. Le mode d'exécution temps virtuel signifie que l'on ne cherche plus à lier le temps réel et le temps virtuel. Chaque cycle est effectué complètement le plus rapidement possible. Ce mode d'exécution constitue bien sûr une infidélité à la réalité virtuelle mais constitue souvent un bon outil de mise au point et de validation. Nous essayons autant que possible de nous passer de ce mode d'exécution.

Les notions de cycle et de pas de temps impliquent que le temps virtuel soit discrétisé. Elles nous imposent également de poser la question de la synchronicité.

4.2.1.3 L'hypothèse de synchronicité

La question de la synchronicité est cruciale lorsqu'on décide d'ordonnancer des objets actifs. Cette hypothèse contraint les activités de perception et d'action des entités autonomes. Pratiquement, nous définissons deux modes d'ordonnancement séquentiel des activités lors

³Dans le domaine des systèmes à temps réel (assez éloigné de la réalité virtuelle), l'acception du terme « temps réel » correspond à ce que nous appelons ici « mode d'exécution temps réel ». L'appellation temps physique est alors exclusivement utilisée pour désigner ce que nous avons appelé temps réel au paragraphe précédent.

d'un cycle : le mode synchrone et le mode asynchrone.

Un ordonnancement synchrone se déroule de la manière suivante :

- perception : lors d'un cycle de simulation toutes les entités autonomes perçoivent l'état du monde de l'instant t ;
- décision : toutes les activités exécutent leurs calculs à partir des perceptions de l'instant t (sur un pas de temps) ;
- action : les entités autonomes mettent à jour les modifications sur leur environnement (le passage de l'instant t à $t+1$).

L'important est donc que toutes les phases de perception soient séparées de toutes les phases d'actions. Il n'y pas de causalité entre les activités à l'intérieur d'un même cycle.

Par contre, lors d'un ordonnancement cyclique asynchrone, l'appel successif des activités conduit à la réalisation successive du cycle indivisible perception-décision-action de chaque entité autonome. Dans ce cas, il existe un lien de causalité entre les différentes activités au sein d'un même cycle de simulation.

Si nous nous fixons comme but un mode d'exécution en temps réel, nous faisons l'hypothèse théorique que les entités autonomes vivent dans un temps continu, le temps physique (ou une fonction idéalement bijective et continue du temps physique). L'activation d'une entité est assimilée à un événement ponctuel dans ce temps physique (le temps d'exécution est négligé). L'hypothèse selon laquelle des événements ponctuels se réalisent exactement au même instant ne se justifie pas.

De plus, l'intérêt du type de simulation que nous proposons est de pouvoir mettre en œuvre des mécanismes telle la compétition entre les phénomènes d'interaction⁴. Avoir un lien de causalité entre les différentes activités au sein d'un même cycle de simulation permet d'assurer la réalisation de ces mécanismes. Cela permet de gérer simplement les conflits⁵ entre deux entités (**Interaction**) partageant les mêmes ressources (**Constituant**). Les solutions classiques permettant de gérer ces conflits en utilisant par exemple, des sémaphores qui modifient les perceptions des entités autonomes ; ce qui introduit une causalité et donc de l'asynchronisme dans un cycle de simulation. Le même problème se pose en ce qui concerne l'action lorsque deux objets **Interaction** décident de modifier différemment le même état d'un **Constituant**, la nouvelle valeur de cet état devient indécidable. A la limite, il est même possible de considérer que l'hypothèse de synchronicité est concrètement peu adaptée puisqu'au final c'est un microprocesseur séquentiel qui effectue les instructions élémentaires d'un programme informatique.

Ensuite, le cycle perception-décision-action est historiquement associé à une posture cognitive. L'hypothèse de synchronicité nous conduit à séparer ces trois temps (voir figure 4.4.a). Notre orientation constructiviste énaïve nous pousse à considérer les activités de perception et d'action comme étant liées et indivisibles, ce que permet la posture asynchrone (voir figure 4.4.b).

Enfin, l'asynchronisme est le mode d'exécution le plus adapté pour l'autonomie. Nous le

⁴Si on s'intéresse à la simulation de réactions chimiques, cela permet par exemple d'assurer la conservation de la matière. Considérons deux réactions R1 et R2, transformant respectivement A en B et A en C. Si lors d'un pas de simulation, la réaction autonome R1 transforme la totalité de la quantité d'espèce A disponible en B, dans le cas synchrone, la réaction R2 ne percevra qu'il n'y a plus de A qu'au cycle suivant. Elle produit donc du C à ce cycle à partir des même perception que R2. Il y aura donc eu création de matière. L'asynchronisme permet d'assurer que $A+B+C$ est constant, même lors d'un cas limite d'utilisation de la méthode. L'erreur numérique affectera les quantités de produits transformés, mais la quantité totale de matière sera conservée.

⁵Les systèmes synchrones présentent des cas d'instabilité qui apparaissent comme des effets de bord. Par exemple, lorsque plusieurs entités autonomes sont redondantes, ces systèmes sont sensibles aux boucles de rétroaction et aux mécanismes de seuils.

constaterons il s'accommode bien d'une modélisation multi-échelles de la dimension temporelle.

Finalement nous observons que l'hypothèse de synchronicité ne se justifie pas théoriquement, qu'elle peut être considérée invalide informatiquement dans certains cas et qu'elle nous paraît inadaptée d'un point de vue cognitif. Ces considérations nous amènent à lever l'hypothèse forte de synchronicité qui est l'apanage du calcul différentiel. Nous choisissons donc un ordonnancement asynchrone pour nos objets actifs **Interaction**. Ceci nous permet d'explorer une voie différente de celle empruntée par nombre d'outils de simulation. Définissons alors plus précisément un premier ordonnanceur.

4.2.2 Ordonnancement asynchrone chaotique sans remise

4.2.2.1 Introduction de la composante stochastique

Le choix de l'ordonnancement asynchrone simplifie la conception du programme informatique. A tout instant du temps physique, il n'existe qu'un unique état du monde : la **Structure** qui est composée de **Constituant**. C'est par la modification de cette structure que les objets **Interaction** s'influencent mutuellement dans un même cycle (causalité).

Cette causalité peut amener une activité à être privilégiée par rapport à une autre si elles sont en permanence appelées dans le même ordre, ce qui risque d'introduire un biais dans la simulation. Afin de pallier ce problème, il est possible de brasser les activités à chaque cycle de simulation. Aucun objet n'est alors privilégié. Le biais éventuellement générée est censée s'annuler sur un grand nombre de cycles de simulation. Ainsi, nous obtenons un ordonnancement que nous qualifions de « chaotique ». Ce brassage, pour être viable, doit assurer un tirage aléatoire équiprobable pour chaque activité. Nous parlons d'ordonnancement sans remise car, durant un cycle chaque activité est appelée une et une seule fois.

Par ce procédé, nous introduisons la composante stochastique dans nos modèles. Nous le verrons, cela constitue un atout majeur du modèle proposé. Pour nombre d'auteurs, cette composante est plus qu'un simple bruit indésirable ; elle est nécessaire au fonctionnement des phénomènes vivants (voir section 1.5). Nos expériences *in virtuo* ne seront donc pas déterministes.

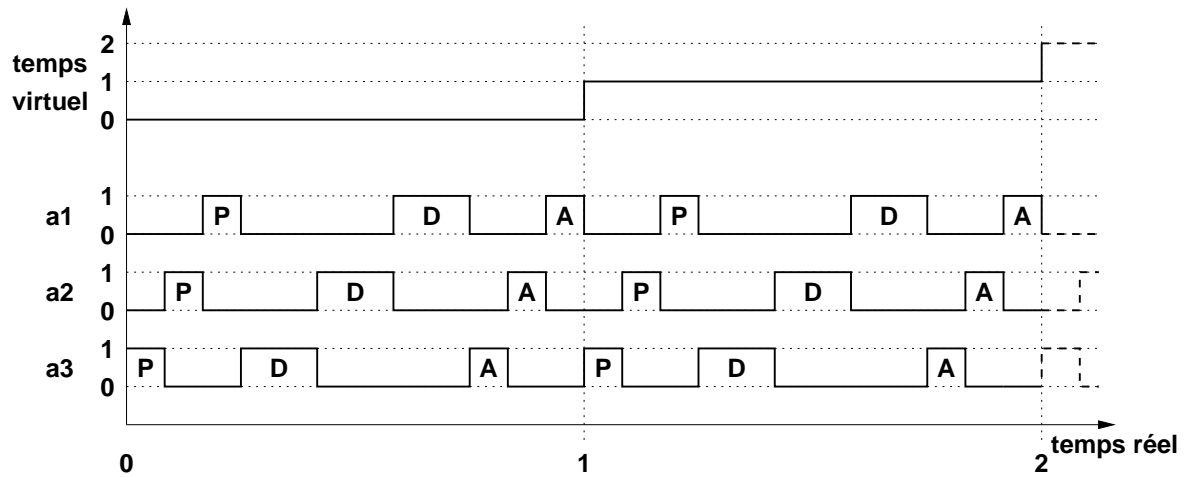
Enfin il faut assurer le mode temps réel de l'ordonnancement chaotique asynchrone sans remise. Pour cela il faut s'assurer que la somme des temps d'exécution de chaque activité est inférieure au pas de temps virtuel choisi. Un compromis entre précision et puissance de calcul est donc à trouver.

Finalement un cycle de simulation correspond à l'exécution de chaque activité suivie d'une phase d'oisiveté pour attendre que le temps réel rattrape le temps virtuel et que l'ordonnanceur lance un nouveau cycle de simulation (voir figure 4.5). [Harrouet, 2001] fournit une description de ce type d'ordonnancement.

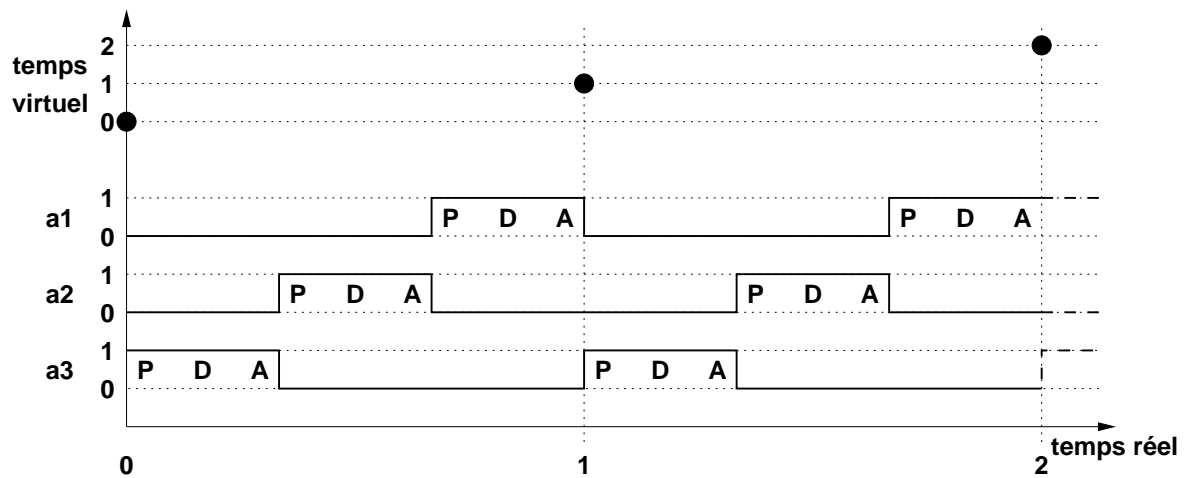
4.2.2.2 Introduction du multi-échelles

La nécessité de trouver un compromis entre la puissance et la précision des calculs, et surtout la volonté de doter nos modèles de propriétés multi-échelles pour la dimension temporelle, nous amènent à enrichir le concept d'ordonnancement des interactions.

En effet, certains objets **Interaction** réalisent des phénomènes qui agissent à des échelles de temps très différentes. Il ne paraît donc pas nécessaire que leur fréquence d'activation soient



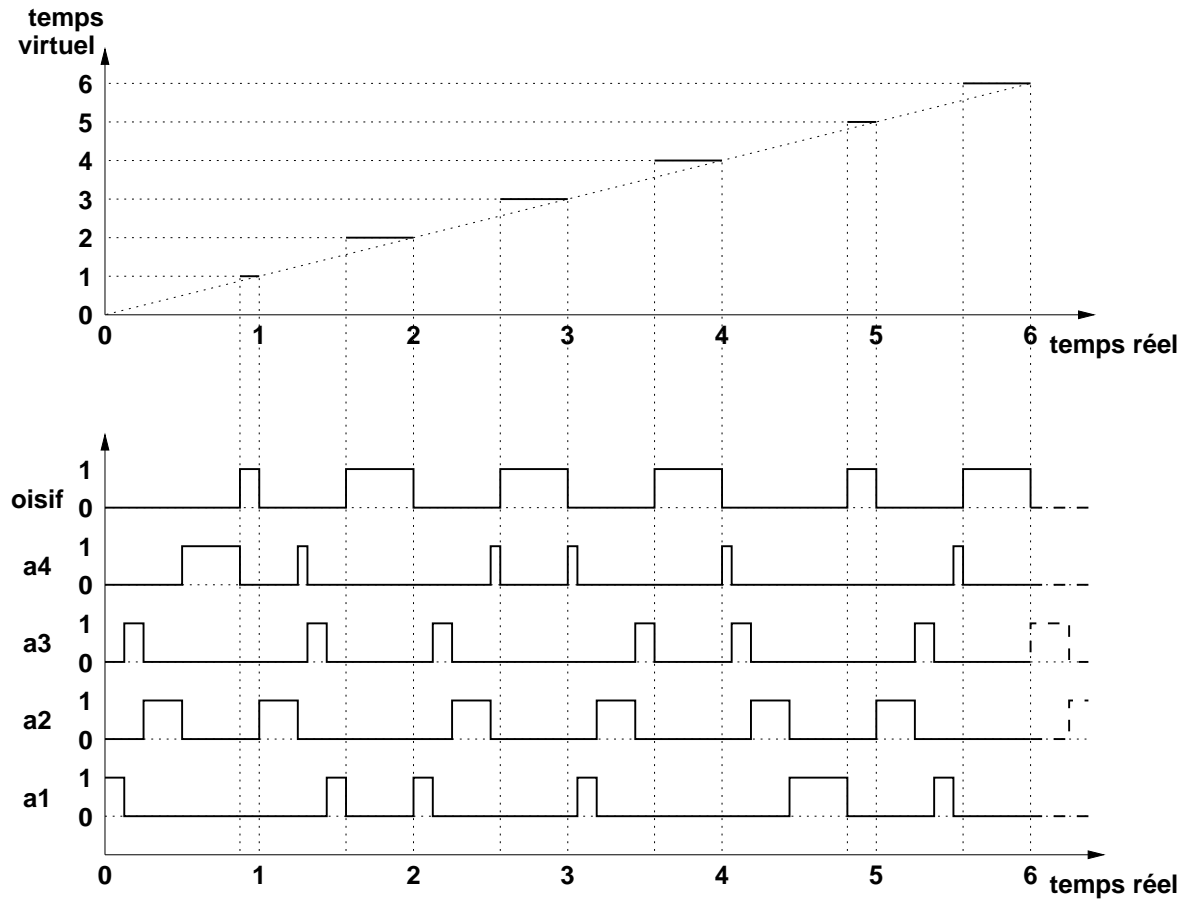
a) Ordonnement synchrone



b) Ordonnement asynchrone

légende a1, a2, a3 : activités P : perception D : décision A : action

FIG. 4.4: Les chronogrammes illustrent les deux types d'ordonnement synchrone et asynchrone des objets actifs. L'ordonnement asynchrone permet de concevoir une activité comme un cycle indivisible perception-décision-action. Remarquons qu'entre deux cycles de simulation asynchrone, nous ne pouvons définir la valeur du temps virtuel.



Ordonnancement asynchrone, chaotique, sans remise

FIG. 4.5: La figure représente l'ordonnancement de quatre activités dans le mode chaotique, asynchrone et sans remise. L'activité oisive, permet à l'application d'attendre que le temps réel rattrape de temps virtuel. Le temps virtuel ne s'exprime en fonction du temps réel que pendant les phases d'oisiveté.

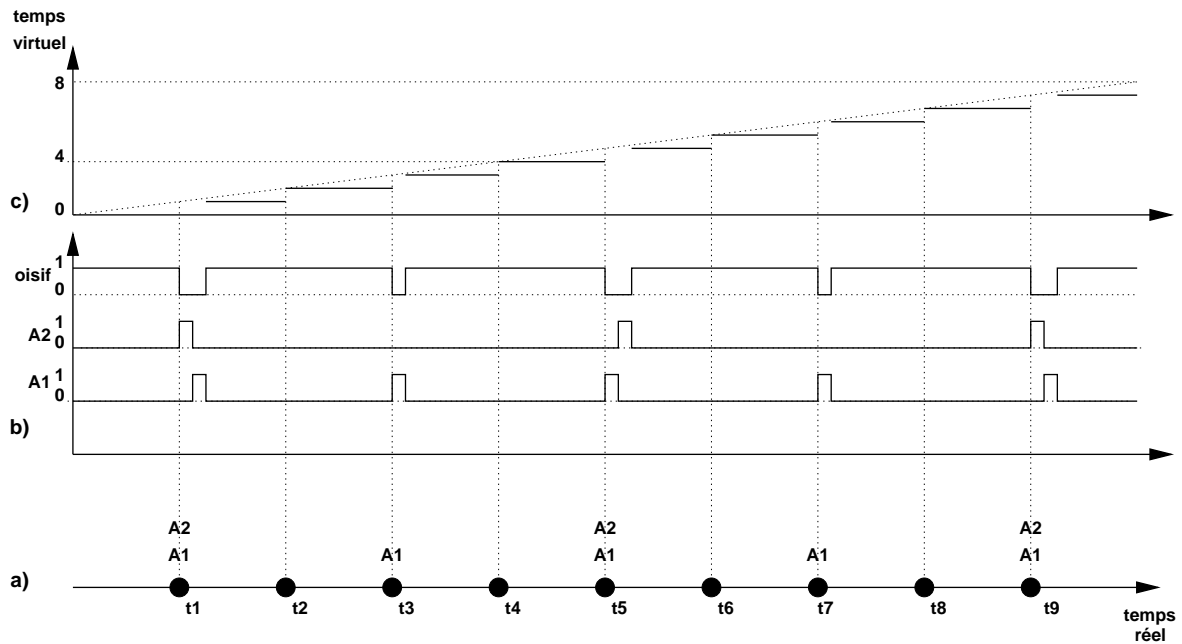


FIG. 4.6: La figure représente l'ordonnement des activités dans le mode chaotique, asynchrone, sans remise et multi-échelles. a) en bas, le temps est découpé en instants. Les appels des activités A1 et A2 sont répartis sur ces instants suivant leur période (2 pour A1 et 4 pour A2). b) Les chronogrammes représentent l'activité des objets autonomes. Lorsque A1 et A2 doivent être exécutées au même instant leur ordre est aléatoire. c) La courbe représente le temps virtuel en fonction du temps réel. Encore une fois le temps virtuel n'est déterminé que pendant les phase d'oisiveté.

les mêmes ; ce qui implique que durant un cycle, toutes les activités ne sont pas forcément activées.

La notion de cycle se transforme alors en instant (voir figure 4.6.a). C'est-à-dire que l'ordonnanceur discrétise⁶ le temps à venir. Les appels des activités sont alors planifiés sur ces instants en fonction de leur période d'activation. Lorsque l'ordonnanceur détecte que le temps réel atteint la valeur d'un instant, il exécute toutes les activités inscrites à cet instant dans un ordre aléatoire (voir figure 4.6.b) afin d'assurer l'équité entre ces activités. Le temps virtuel évolue alors de la même manière que précédemment (voir figure 4.6.c). À la différence de ce qui se passait dans la version précédente, on n'attend plus que le temps réel rattrape le temps virtuel mais on déclenche les calculs lorsqu'ils doivent être exécutés ; ce qui introduit une certaine latence.

4.2.3 Discussion

4.2.3.1 Asynchrone synchronisé ?

Pour l'ordonnanceur, l'exécution d'une activité associée à un objet **Interaction** est considérée comme un événement ponctuel (à une latence près), intervenant avec une certaine récurrence. Cet événement permet à l'objet **Interaction** de passer de l'instant t à l'instant $t + 1$, tout en modifiant la structure à laquelle il est associé. Nous avons ici un ordonnancement à événements discrets d'activités modélisant des phénomènes agissant sur

⁶L'intervalle entre les instants doit être plus court ou de même durée que la plus petite période d'activation dans le système.

des intervalles de temps. C'est ce principe qui permet d'introduire cette notion de causalité entre événements et de se passer de l'hypothèse de synchronicité.

Tout se passe comme si l'objet autonome **Interaction** possède sa propre définition du temps virtuel qu'il fait évoluer au fil de ses activations. La figure 4.5 a fait apparaître l'expression du temps virtuel global dans le système en fonction du temps réel. Nous nous sommes permis de représenter ce temps global seulement pendant les phases d'oisiveté, c'est-à-dire pendant les périodes du temps réel ou les hypothétiques temps virtuels individuels des entités autonomes sont à la même valeur. Deux remarques nous viennent alors.

Premièrement, les activations d'objet, considérées comme des actions instantanées dans le temps réel, réalisent des actions qui correspondent à des intervalles dans le temps simulé. Nous pouvons considérer que la notion de cycle permet aux entités autonomes de synchroniser leurs « temps virtuels individuels ». Pendant un cycle, on attend que tout le monde soit passé de t à $t + 1$, puis après une phase d'oisiveté, on relance un nouveau cycle. L'ordonnancement que nous appelons « chaotique asynchrone et sans remise » n'est asynchrone que par rapport au temps physique. Il présente cependant une forme de synchronisme dans le temps simulé.

Deuxièmement, lorsque les activités ont des périodes d'occurrence différentes, si nous choisissons de définir la valeur du temps virtuel global seulement quand les temps individuels de chaque entité autonome ont la même valeur, alors la définition du temps virtuel global est erronée pour les temps $t3$, $t4$, $t7$ et $t8$ de la figure 4.6. En effet à ces temps les objets autonomes n'ont pas atteint le même temps de simulation. Le temps virtuel est valide seulement lorsqu'un instant t correspond à un commun multiple des périodes. Or, dans le cas général, rien ne garantit l'existence d'un plus petit multiple. Ce constat, nous indique qu'il est impossible de définir un temps virtuel global avec notre mode d'ordonnancement.

Est-ce important ?

À chaque activation, un objet autonome incrémente son horloge interne de la durée d'une période. Ainsi il n'a jamais besoin de consulter d'autres valeurs du temps que celle qu'il se définit lui-même. Le rôle de l'ordonnanceur n'est alors plus de garantir la cohérence du temps virtuel global, mais de garantir la cohérence des temps propres à chaque entité autonome. Le temps virtuel global n'est alors plus utilisé⁷, il n'est donc plus nécessaire de pouvoir le définir.

La notion de cycle avait pour fonction de permettre la définition d'un temps global. Si nous considérons que cette possibilité contraint inutilement le mode d'ordonnancement, il est possible de ne plus utiliser cette notion de cycle.

4.2.3.2 Ordonnancement avec remise

Nous l'avons déjà dit, pour l'ordonnanceur, l'ordonnancement correspond à la planification et l'exécution d'événements supposés discrets, ces événements ayant une certaine fréquence d'occurrence. Il est possible de transformer la notion de fréquence d'activation en probabilité d'occurrence en fonction du temps. Dans ce cas, nous passons d'un ordonnancement chaotique sans remise à un ordonnancement avec remise.

L'algorithme de Gillespie [Gillespie, 1976] est une solution d'ordonnancement asynchrone chaotique avec remise d'événements discrets. Le principe de cet algorithme est illustré par la figure 4.7.

Il est adapté pour planifier l'exécution des activités pas à pas. Le problème de la version

⁷Si l'utilisateur désire l'estimer, il peut regarder sa montre qui lui indique son temps individuel.

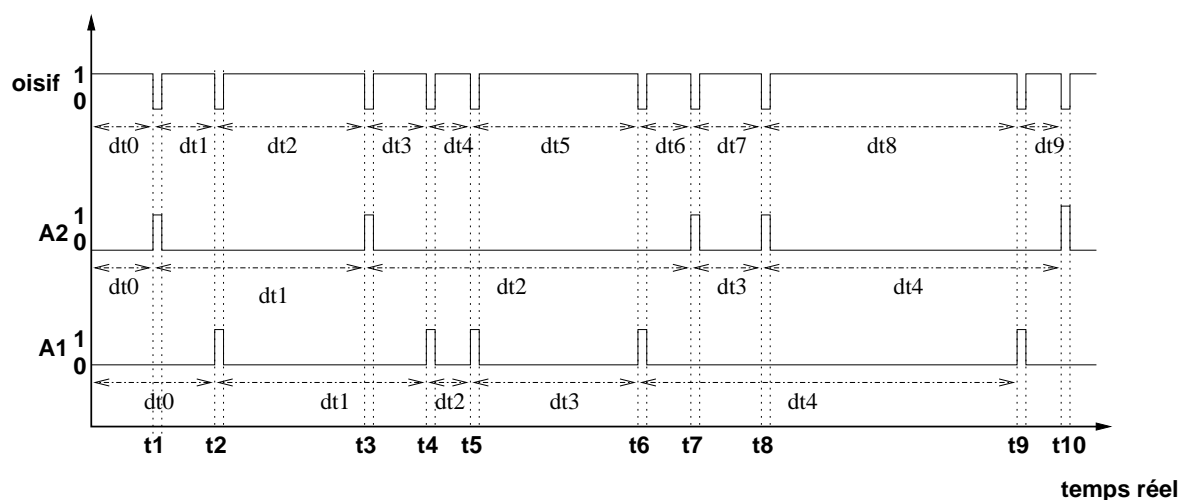


FIG. 4.7: Les chronogrammes illustrent comment l'algorithme de Gillespie[Gillespie, 1976] peut ordonnancer deux activités A1 et A2. À chaque pas de simulation, en fonction des différentes probabilités d'occurrence, on tire une variable aléatoire permettant de déterminer l'activité à exécuter, puis on tire un temps à patienter avant la réalisation du pas suivant (le dt durant lequel la tâche oisive est active). À chaque appel d'une activité, l'ordonnanceur indique l'intervalle de temps à simuler à l'activité (le dt des tâches A1 et A2).

classique de l'algorithme de Gillespie est que le nombre de phénomènes à ordonnancer ne varie pas au cours des calculs. Pour réaliser cet ordonnancement dans le mode temps réel, il faut réfléchir au calcul des variables aléatoires et à la prise en compte de modifications imprévues du système. Il faut également penser à prendre en compte la latence due au temps d'exécution de l'activité.

Dans ce mode d'ordonnancement, le temps est alors une variable interne de l'ordonnanceur, accessible par lui uniquement. Chaque entité connaît son temps par l'histoire des appels de son activité. L'ordonnanceur n'étant pas à proprement parler dans l'univers virtuel, tout se passe comme s'il n'y avait plus de temps global dans le système, mais que chaque entité autonome du monde virtuel possédait son propre temps. L'utilisateur que l'on considérera comme un modèle parmi d'autres pourra se référer à son propre temps, le temps physique, pour expérimenter le système. Cette vision pourrait se révéler intéressante pour répartir les calculs sur plusieurs machines. Elle permettrait d'entrevoir une simulation au niveau mésoscopique (voir section 2.3.5.3). Il serait également possible de faire le lien avec les travaux autour de l'algèbre de processus (voir section 2.2.2). Le verrou technique réside dans l'optimisation du calcul des variables aléatoires.

4.2.4 Solution technique adoptée

Nous avons fait apparaître deux types d'ordonnancement pour gérer nos interactions réifiées :

- chaotique, asynchrone, avec remise,
- chaotique, asynchrone, sans remise.

Après discussion, ce dernier apparaît comme une demi-mesure par rapport à notre volonté d'appliquer un ordonnancement asynchrone multi-échelles. Cependant, les travaux du Centre Européen de Réalité Virtuelle repose sur l'usage de la bibliothèque ARéVi [Harrouet et al., 2006] qui fournit un ordonnancement chaotique asynchrone sans remise. L'ordonnan-

Le problème se pose « moins » si nous choisissons de manière adéquate les périodes des interactions.

La solution technique adoptée n'est donc pas idéale. Elle apparaît comme une solution temporaire laissant des perspectives d'évolution à notre travail.

Dans ce chapitre, nous avons posé les bases du modèle : les interactions. Nous avons décrit comment elles peuvent être ordonnancées : ordonnancement chaotique et asynchrone. Nous devons maintenant décrire comment apparaissent et disparaissent ces interactions.

4.3 Organisations multi-niveaux

4.3.1 L'organisation

Il est intéressant de réifier les relations entre les composants pour reproduire la dynamique d'un système complexe, mais cela ne suffit pas à rendre intelligible son fonctionnement. Utiliser la notion d'organisation est le moyen de mettre en œuvre la modélisation systémique et de rendre intelligible cette modélisation [Le Moigne, 1999]. C'est d'ailleurs un concept central des biologies systémique, intégrative ou théorique. Nous souhaitons donc lier cette notion d'organisation à notre principe d'autonomie et aux interactions réifiées.

Il existe déjà de nombreux modèles d'organisation pour la conception de systèmes multi-agents : OMNI [Javier Vázquez-Salceda, 2004] (Organizational Model for Normative Institutions), MOISE+ [Hübner and Sichman, 2002] (Model of Organization for multi-agent Systems), AGR [Ferber et al., 2004] (Agent Groupe Rôle), MASCARET [Querrec, 2002; Buche et al., 2004] (MultiAgent System for Collaborative Adaptive Realistic Environment for Training), RIO [Seck, 2003] (Rôle Interaction Organisation). Les deux derniers s'intéressent même à la réification des interactions. Malheureusement, ces modèles sont orientés organisations sociales et ne nous sont pas d'une grande utilité. Ils utilisent en effet une approche qui est destinée à permettre la reproduction de schémas organisationnels bien connus et associés à un but. Or, dans le cadre de l'étude des systèmes complexes vivants, nous ne connaissons pas encore précisément ces schémas organisationnels, c'est d'ailleurs un des objectifs de l'étude. Ainsi, la notion de rôle présente dans ces modèles, ne nous paraît pas adaptée dans ce cas. Il nous faut définir un modèle plus adapté à la biologie.

La notion d'organisation en biologie a été introduite dans les travaux du CERV par Gabriel Querrec. La thèse [Querrec, 2002] replace cette notion d'organisation dans le contexte de la biologie systémique. Bien qu'il s'agisse d'une notion fondamentale en biologie, il demeure difficile d'en trouver une définition communément admise. C'est pourquoi nous la définissons simplement, pour permettre au modélisateur de s'adapter suivant ses orientations épistémologiques.

Trois étapes permettent alors de construire et d'intégrer l'organisation dans notre modèle générique. Nous la définissons tout d'abord en tant qu'état. Nous nous intéressons ensuite à sa dynamique pour introduire la notion de phénomène. Enfin, nous montrons comment les organisations qui désignent à la fois états et processus, peuvent être agencées pour modéliser un système complexe.

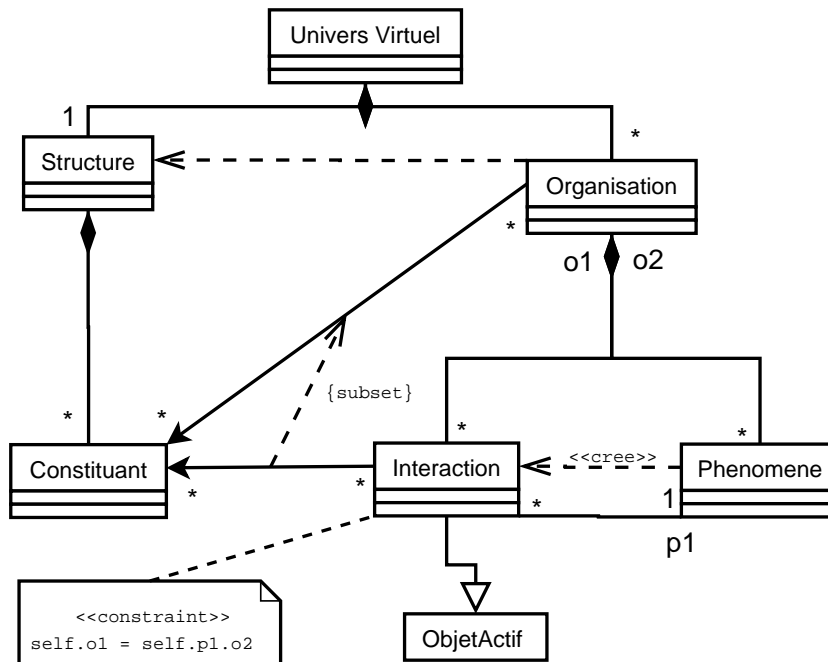


FIG. 4.10: Le diagramme UML montre l'intégration de la classe **Phénomène** dans le modèle générique. Un phénomène accède aux objets **Constituants** via son organisation. Il est l'objet qui crée les objets **Interaction**.

qui n'agit pas directement sur sa structure, mais qui instancie les interactions agissant dessus.

Un objet **Phénomène** a accès à tous les constituants qui sont inclus dans la topologie de l'organisation à laquelle il appartient. Son rôle est de détecter les situations qui nécessitent l'émergence de nouvelles interactions. Il s'agit là d'une tâche difficile à mettre en pratique. En effet, la solution idéale serait que chaque phénomène scrute en permanence l'état de la structure. Malheureusement, la taille des simulations envisagées ne permet pas de pratiquer l'attente active pour tous les phénomènes. De ce fait, chaque phénomène suivant sa nature encapsule un mécanisme de détection. Plusieurs mécanismes sont alors à envisager.

- **l'attente active.** C'est la pire des solutions d'un point de vue efficacité. Le phénomène est un objet actif qui à chaque appel de son activité vérifie toute la structure de l'organisation ;
- **l'attente passive événementielle sur la topologie.** Le phénomène réagit aux modifications de la topologie de la structure de l'organisation, c'est-à-dire à chaque ajout ou suppression d'un constituant ;
- Une attente **mixte** « mélange » des deux précédentes. Le phénomène construit la liste des quelques constituants susceptibles de l'intéresser et procède à une attente active sur ces quelques objets. La liste est alors mise à jour de manière événementielle lorsque la topologie change ;
- **l'attente passive événementielle sur les constituants.** Le phénomène s'inscrit à un mécanisme d'événements pour détecter la modification de l'état des constituants qui l'intéresse.

Finalement toutes les combinaisons de ces mécanismes sont possibles. Les mécanismes

événementiels sont préférables¹⁰, car moins coûteux en temps de calcul. Mais, ils doivent toutefois être manipulés avec précaution pour ne pas dénaturer le caractère autonome des modèles. Ce sont des optimisations locales dans le modèle informatique.

L'avantage de l'architecture utilisant les phénomènes est qu'elle permet la modularité. Ainsi, les mécanismes d'optimisation sont définis localement et ne contraignent pas le reste de l'application. Les mécanismes spécifiques à la détection d'un phénomène sont codés dans la classe **Phénomène** uniquement. Les mécanismes permettant la réalisation de ces phénomènes sont eux définis dans la classe **Interaction**. Pour construire une organisation, le modélisateur doit définir une topologie et les phénomènes. Les interactions émergeront et vivront de manière autonome sans que le modélisateur n'ait à les instancier.

4.3.4 Les organisations

Le distinction entre un système et son environnement est un choix arbitraire et subjectif de la part du modélisateur. Pour rendre un système complexe intelligible, le modélisateur devra définir et concevoir le système global comme un ensemble de systèmes en interaction. Étant donné que notre modèle utilise une structure globale représentant l'état du monde, la décomposition du système global passe par la définition de plusieurs organisations portant sur des éléments de la structure. Le système global est alors interprété comme un agencement d'organisations qui font évoluer l'état du système.

La réification des interactions facilite la mise en œuvre d'une modélisation du vivant par systèmes autonomes en couplage structurel, ainsi que le propose Varela dans l'ouvrage¹¹ « autonomie et connaissance » [Varela, 1989a]. En effet, un système est un ensemble de constituants et d'interactions associés à une organisation. Lorsque deux systèmes ont certains éléments de structure en commun, c'est-à-dire que les deux organisations agissent sur des mêmes constituants, on dit qu'ils sont en couplage structurel (voir figure 4.11). Ils perçoivent chacun l'influence de l'autre système comme une perturbation sur leur propre structure, sans se connaître l'un l'autre. La réification des interactions nous ayant poussés à considérer la structure isolément comme une somme d'objets passifs représentant simplement des états, le couplage structurel entre les organisations s'obtient immédiatement. De plus, les interactions étaient déjà elles-mêmes en couplage structurel les unes avec les autres par l'intermédiaire des constituants. Rappelons que le singleton **Structure** nous fait penser au tableau noir d'une architecture en *blackboard* (voir section 2.4.5).

De plus, la délimitation de la structure d'une organisation (ou d'un système) est choisie arbitrairement et subjectivement par le modélisateur. Suivant l'évolution du système, elle peut être amenée à changer. Par exemple si un nouveau constituant apparaît dans l'univers virtuel et que ce constituant possède les caractéristiques pour être inséré dans la structure de l'organisation¹², un mécanisme doit être présent pour mettre à jour la topologie de l'organisation. Compte tenu de notre principe d'autonomie, nous avons décidé que ce mécanisme devait se trouver dans la classe **Organisation**. Ce mécanisme étant lié à une décision subjective du modélisateur, il est impossible de le définir à coup sûr de manière

¹⁰Notons que dans l'ensemble des applications réalisées à ce jour, nous avons toujours réussi à nous passer de la solution de l'attente active.

¹¹Ce livre est d'ailleurs une des sources d'inspiration principale du modèle générique que nous proposons dans cette thèse. Les travaux exposés dans ce livre précédent également l'établissement par le même auteur du modèle de l'énaction présenté dans l'état de l'art.

¹²Par exemple, si nous avons une organisation « cellule » et une organisation « milieu ». Lors de l'exocytose, des constituants « protéine » passeront de l'organisation « cellule » à l'organisation « milieu ».

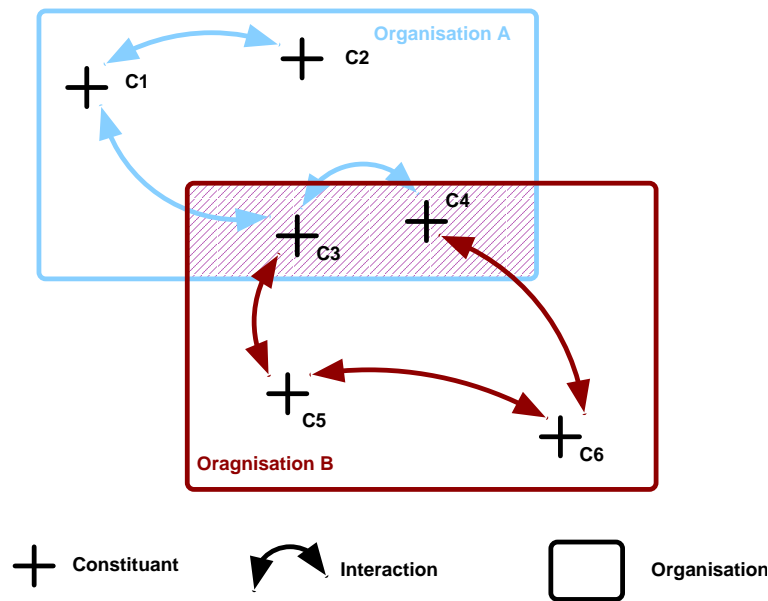


FIG. 4.11: La figure nous montre deux organisations A et B en couplage structurel. L'organisation A a pour structure l'ensemble des constituants $\{C1, C2, C3, C4\}$ et l'organisation B $\{C3, C4, C5, C6\}$. L'ensemble $\{C3, C4\}$ représente la zone de couplage structurel entre A et B. Les deux systèmes correspondant aux deux organisations s'influenceront mutuellement en modifiant l'état de C3 et C4.

générique. Cependant, nous pouvons le circonscrire en choisissant d'impliquer seulement les classes **Organisation** et **Structure**. Ainsi, le choix se trouve être de même nature que celui évoqué pour les phénomènes. Une gestion de la topologie par attente active est à éviter, alors qu'un système plus passif, événementiel est plus efficace. Pour se faire, l'organisation peut s'inscrire auprès du singleton **Structure** et demander à recevoir un événement à chaque création d'un constituant d'un certain type. L'expérience de modélisation que nous avons eue jusqu'à aujourd'hui, montre qu'en général assez peu d'organisations ont une topologie variable et que lorsque cela arrive, le mécanisme événementiel¹³ suffit.

Le rôle opérationnel des interactions (modification de l'état du monde), des phénomènes (création des interactions) et de l'organisation (gestion de la topologie) permettent de doter le système particulier associé à chaque organisation des moyens et des conditions de la génération et de la réalisation de ses propres processus. Ainsi nous manipulons des systèmes opérationnellement clos. Pour Varela, cette clôture opérationnelle et la construction par couplage structurel impliquent que les systèmes manipulés sont des systèmes autonomes.

Finalement cette approche permet l'établissement d'un modèle par la juxtaposition de systèmes autonomes. Constatons au passage que nous n'avons pas réifié le concept de système dans notre architecture informatique. En effet l'objet **Organisation** suffit à manipuler les systèmes dans la mesure où la seule différence entre un système et une organisation (au sens où nous la définissons) est le lien avec les constituants : l'organisation est associée aux constituants ; le système est composé des constituants. Une classe **Système** n'aurait rien apporté de plus au modèle. Pour enrichir les possibilités de modélisation du modèle générique, nous avons permis une composition hiérarchique des organisations (donc des systèmes). Ainsi, en plus d'être composé d'objet **Interaction** et **Phénomène**, l'objet **Organisation** peut se

¹³La bibliothèque ARéVi fournit un mécanisme d'envoi d'événements. Un système d'abonnement permet à un objet d'être averti à chaque occurrence de l'événement pour lequel il est abonné.

mère et nous ramène au premier cas.

Chaque classe du modèle générique possède ainsi des méthodes permettant la mise à jour du système à chaque modification intervenant dans le modèle applicatif.

Remarquons enfin que l'interaction et l'organisation pourraient être considérées comme étant de même nature. Dans l'arbre de décomposition d'une organisation, les interactions en sont les feuilles. L'organisation possède toutes les propriétés d'une interaction. L'organisation serait finalement définie comme une interaction complexe de second ordre composée par d'autres interactions.

Nous avons dans ce chapitre défini un modèle générique à partir de cinq classes de base. Ce modèle est un cadre de modélisation qui est :

- 1) **multi-modèles** : grâce à la réification des interactions en entités autonomes, car chaque interaction peut traduire des phénomènes de natures complètement différentes.
- 2) **multi-échelles temporelles** : mis en œuvre par l'utilisation du principe de l'ordonnement temps réel, chaotique, asynchrone avec ou sans remise.
- 3) **modulaire** : l'agencement des organisations représentant des systèmes autonomes en couplage structurel.

Il nous reste maintenant à indiquer comment il est possible de dériver ce modèle abstrait pour modéliser le vivant. Cette tâche nous éclairera sûrement un peu plus quant à la mise en œuvre de cette architecture.

Chapitre 5

Dérivation du modèle pour la biologie

Telles que nous les avons définies, les classes `Constituant`, `Interaction` et `Phénomène` sont des classes abstraites. Elles doivent être dérivées et spécialisées suivant le type de phénomène que l'on souhaite mettre en œuvre. Au fil des travaux menés par l'équipe EBV, certains phénomènes sont intervenus de manière récurrente. Dans ce chapitre, nous décrivons comment ces phénomènes peuvent être modélisés dans le cadre que nous avons fixé. Ainsi nous évoquons quatre natures différentes de modèles. La première section présente les phénomènes chimiques. Ensuite, nous exposons l'organisation de la cellule. Nous montrons également comment les phénomènes mécaniques peuvent être décrits. Le dernier modèle présenté est un peu particulier car nous nous intéressons à la modélisation de l'utilisateur dans l'univers virtuel. Pour finir, nous présentons la librairie qui implémente tous ces modèles.

5.1 Les phénomènes chimiques

La première catégorie de phénomènes à modéliser est celle des phénomènes chimiques. C'est la première et souvent la seule nature de phénomène prise en compte dans l'étude des systèmes complexes vivants. L'histoire de nos applications nous a amenés à utiliser une modélisation macroscopique des réactions chimiques. Ainsi nous nous appuyons sur les agents `Réaction` présentés à la section 3.4.3. Deux types d'interaction sont alors à envisager : temporel et spatio-temporel. Nous présentons donc, respectivement, les réactions et la diffusion. Ensuite, afin de faciliter la définition des modèles, nous avons intégré la norme SBML dans notre modèle. Enfin, nous indiquons comment il est possible de décrire un milieu chimique maillé.

5.1.1 L'interaction Réaction

Pour modéliser les réactions chimiques au niveau macroscopique, nous nous appuyons sur les agents `Réaction`. L'effort à fournir pour les adapter à notre modèle est minime puisque ce travail tend à les généraliser. Cependant, quelques notions ont été ajoutées pour assurer le multi-modèles. Les agents `Réaction` manipulaient un tableau de concentrations ; nos interactions `Réaction` agissent sur les constituants `Espèce` et `Compartment` (voir figure 5.1). L'espèce représente la population de molécules présente dans le milieu homogène. La classe `Espèce` possède donc un attribut « quantité » (nombre de moles). L'objet `Compartment` représente le milieu homogène dans lequel se trouve la population. Il a donc au moins pour

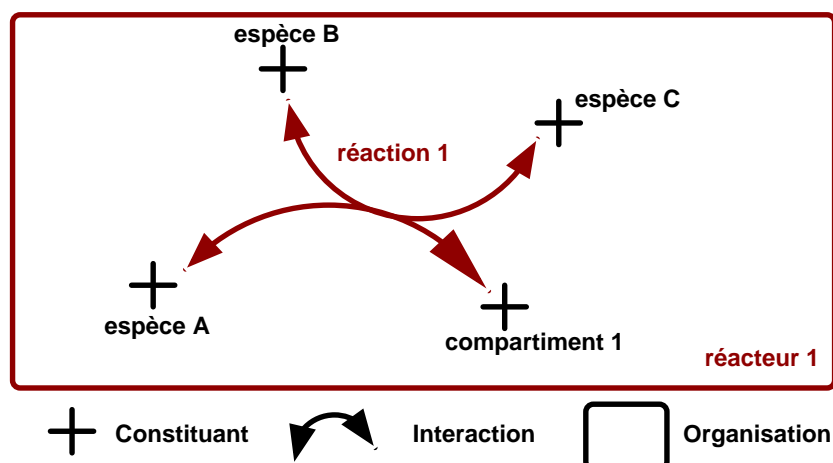


FIG. 5.1: L'organisation **réacteur 1** a pour ensemble structurel les constituants **espèce A**, **B** et **C**, et **compartiment 1**. Les espèces représentent des quantités (nombre de moles). Le compartiment représente la forme, le volume du réacteur. L'interaction **réaction 1** est du type : $A + B \rightarrow C$. Elle lie les trois espèces ainsi que le compartiment dont l'état affecte les calculs (car la vitesse dépend de la concentration qui dépend elle-même du volume).

propriété un volume¹ qui permet à la réaction de calculer des concentrations d'espèces.

Finalement l'objet actif **Réaction** est une réaction :

- qui est couplée aux constituants **Espèce** et **Compartiment** ;
- qui perçoit les quantités des espèces chimiques et le volume du compartiment ;
- qui calcule en fonction d'une loi cinétique sa vitesse de réaction, et l'intègre sur les pas de temps donnés par l'ordonnanceur ;
- qui modifie les quantités des espèces mises en jeu pour réaliser effectivement la réaction chimique.

Dans notre implémentation, nous avons défini une simple méthode d'Euler (voir section 2.1.1). Il est bien sûr possible d'améliorer le mécanisme de décision utilisé par l'activité de la réaction pour améliorer la précision des calculs.

Pour respecter la logique de notre modèle, le modélisateur ne doit pas créer explicitement les réactions, mais doit créer un objet de type **RéactionPhénomène** qui est en charge de détecter si la réaction doit exister. C'est-à-dire qu'il vérifie la présence des espèces qui interviennent dans la réaction et l'instancie au besoin. Nous appelons **Réacteur** l'organisation qui connaît le compartiment, les espèces, et qui contient les phénomènes décrivant les réactions et les réactions elles-mêmes (voir figure 5.2).

Enfin, bien que nous ayons enrichi certaines parties du modèle d'agents **Réaction** pour l'insérer dans notre modèle, l'exécution reste strictement équivalente à celle définie originellement. La validité de l'outil est donc assurée de la même manière qu'elle l'était dans la version originale. Indiquons seulement que la convergence des calculs a été étudiée dans le cas d'un pas fixe pour toutes les réactions ; nous avons utilisé ce principe dans nos applications. Il serait également intéressant d'étudier la convergence dans le cas du multi-échelles temporel.

¹La présence du constituant **Compartiment** évite ainsi la redondance des informations le concernant, ce qui aurait été le cas si les constituants **Espèce** avaient possédé une information sur la concentration au lieu de la quantité. Il devient alors possible par un simple changement de l'attribut volume (et pas une seule instruction de plus), d'en observer l'influence sur les phénomènes chimiques qui s'y déroulent.

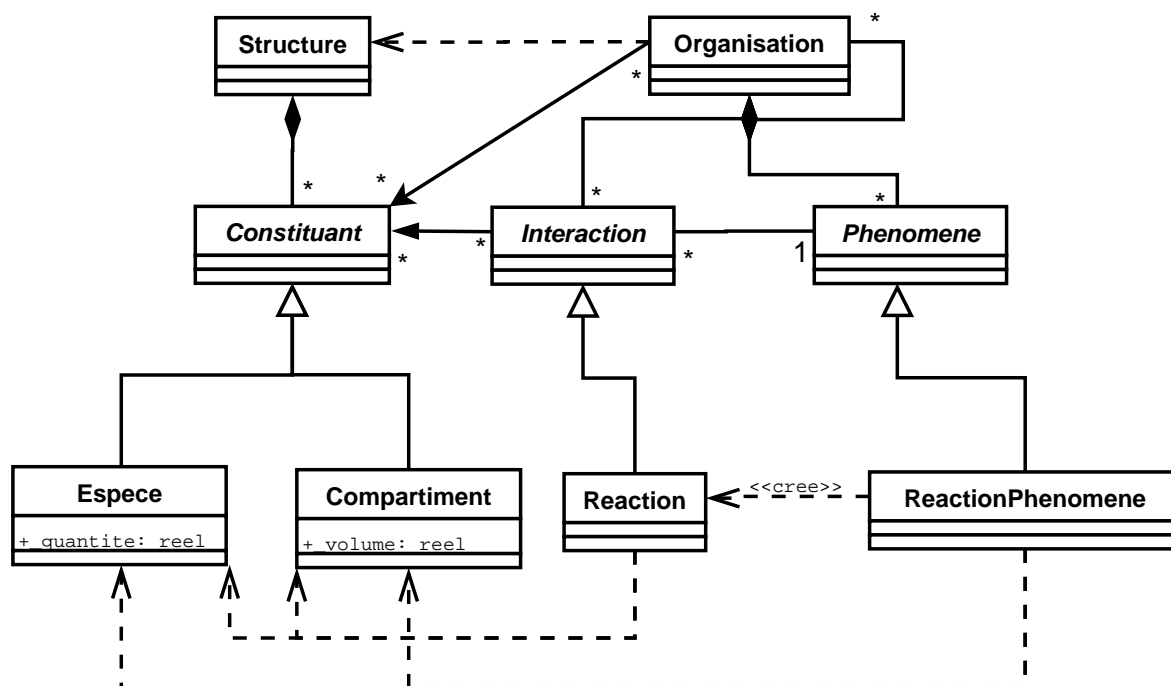


FIG. 5.2: Le diagramme UML représente l'héritage des classes permettant la réalisation des réactions chimiques.

5.1.2 L'interaction Diffusion

Le principe de la diffusion est sensiblement le même que celui des réactions chimiques, à l'exception près qu'il fait intervenir la dimension spatiale. Au niveau macrosopique, la diffusion peut être considérée comme un phénomène de transport d'une portion de la population d'une espèce d'un réacteur chimique à un autre (voir figure 5.3)

Une instance de la classe `DiffusionPhénomène` est ajoutée à l'organisation qui contient les réacteurs. Lorsque les compartiments des réacteurs représentent des parties de l'espace connexes, alors il y a une surface d'échange à travers laquelle peut se réaliser la diffusion moléculaire. L'objet actif `Diffusion` est une interaction :

- qui est couplée aux constituants `Espèce` et `Compartiment`
- qui perçoit la quantité des espèces chimiques, les volumes et les positions des compartiments connexes ;
- qui calcule en fonction de la loi de Fick sa vitesse de diffusion, et l'intègre sur les pas de temps donnés par l'ordonnanceur ;
- qui modifie les quantités des espèces mises en jeu pour réaliser la diffusion moléculaire.

Dans un premier temps nous considérons que les compartiments sont des cubes, ce qui simplifie les calculs. La quantité Q de molécules à passer d'une maille à l'autre lors d'un intervalle est :

$$Q = D * dt * coté * ([EspèceA] - [EspèceB])$$

Remarquons que la différence de nature entre la réaction chimique (transformation) et la diffusion (transport) n'affecte en rien l'intégration de cette dernière dans le modèle. Informatiquement, ce sont pratiquement les mêmes classes. Les seules différences résident dans la méthode du phénomène qui teste la condition de création de l'interaction, et dans l'activité

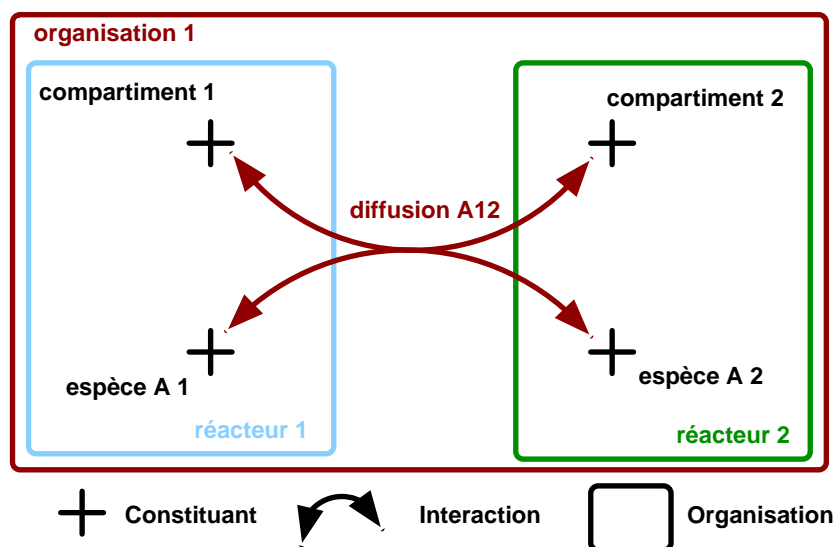


FIG. 5.3: L'interaction diffusion A12 réalise le transport de quantité de matière du constituant *espèce A 1* au constituant *espèce A 2* (où *Espèce A i* correspond à l'espèce A dans le réacteur i). Le calcul des quantités de matière transportée dépend également de la forme du volume et surtout de l'interface entre *compartment1* et *compartment2*.

de l'interaction qui calcule les quantités de produit modifié. Le diagramme de la figure 5.4 est donc pratiquement identique au précédent (figure 5.2).

5.1.3 L'organisation modèle SBML

Finalement, il est possible de décrire n'importe quel réseau biochimique grâce aux modèles de réaction et de diffusion. Pour simplifier la définition des réseaux, nous avons choisi d'utiliser la norme SBML (voir section 2.4.2). Ainsi, il devient possible d'utiliser les différents éditeurs graphiques associés à la norme. Il est également possible d'importer depuis les bases de données SBML, les différents modèles établis par d'autres chercheurs.

L'organisation `ModèleSBML` possède un interpréteur XML qui traduit les différentes balises SBML en objet de notre modèle (voir figure 5.5). L'organisation `ModèleSBML` hérite de la classe `Réacteur`. Pour chaque balise `<compartment>`, elle instancie un constituant `Compartment`. Pour chaque balise `<species>`, elle instancie un constituant `Espèce`. Et pour chaque balise `<reaction>`, elle instancie un phénomène `RéactionPhénomène`. Les réactions sont instanciées par les phénomènes lors de la simulation. Afin d'augmenter les performances des réactions, un code C++ en lien avec chaque balise `<kineticLaw>` est généré et compilé dynamiquement. Les bibliothèques dynamiques générées correspondent à la phase de décision des réactions. Le code généré détermine également la méthode de résolution numérique (Euler dans notre cas).

Outre le fait d'optimiser le temps de traitement des activités des réactions, ce mécanisme permet de modifier dynamiquement les lois cinétiques des réactions. En effet, il suffit de modifier le code généré, de le compiler et de le recharger à la volée. Cette action pouvant intervenir à tout moment de la simulation ; elle peut être utile pour modifier le modèle en cours d'exécution, réalisant ainsi la médiation de l'esprit.

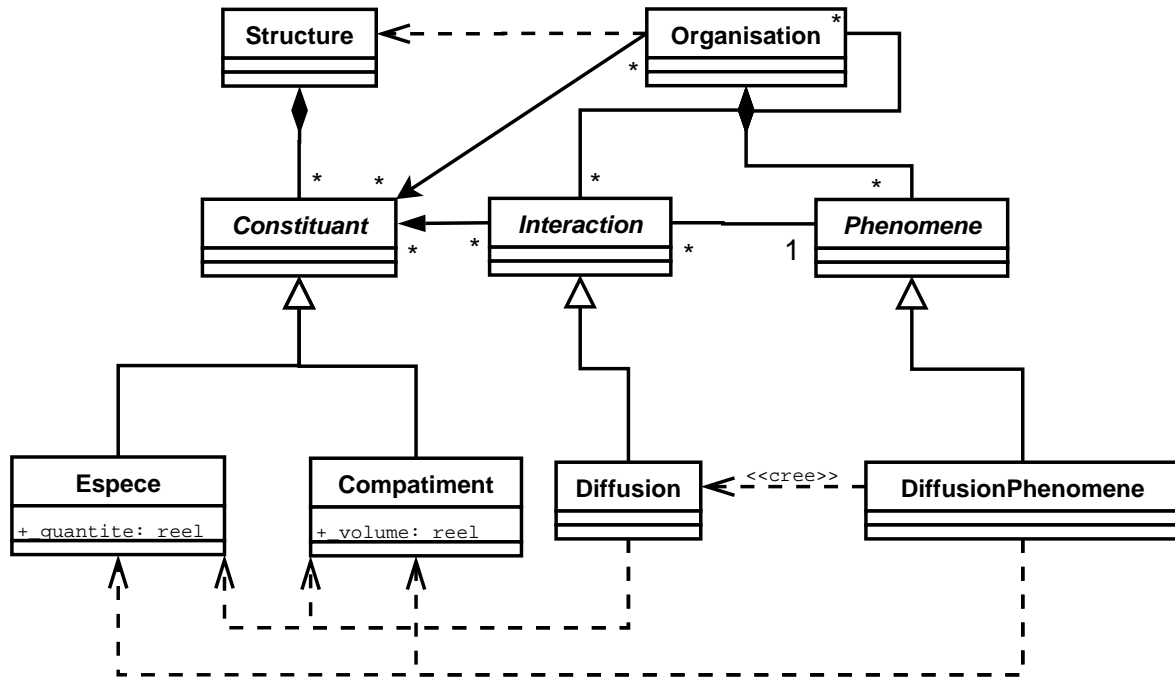


FIG. 5.4: Intégration de l'interaction et du phénomène de la diffusion.

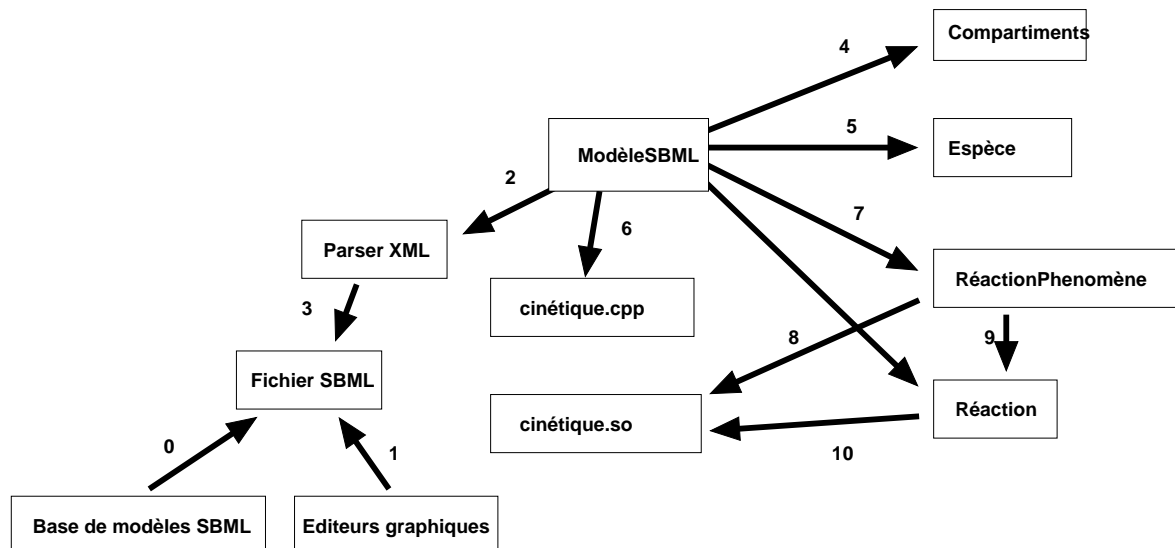


FIG. 5.5: Le modélisateur récupère un modèle SBML sur une base de données (0) ou génère le sien avec un éditeur (1). Ensuite, lors de sa création, un objet `ModèleSBML` utilise un interpréteur XML (2) pour traduire le fichier SBML (3) en compartiments (4), espèces (5). Il génère le code C++ correspondant aux réactions chimiques(6) et crée les phénomènes de réaction (7). Les phénomènes créent les bibliothèques partagées en compilant le code généré(8) puis créent les interactions (9), les conditions d'apparition étant normalement satisfaites. Lors de sa création, la réaction charge la librairie permettant son activité (10).

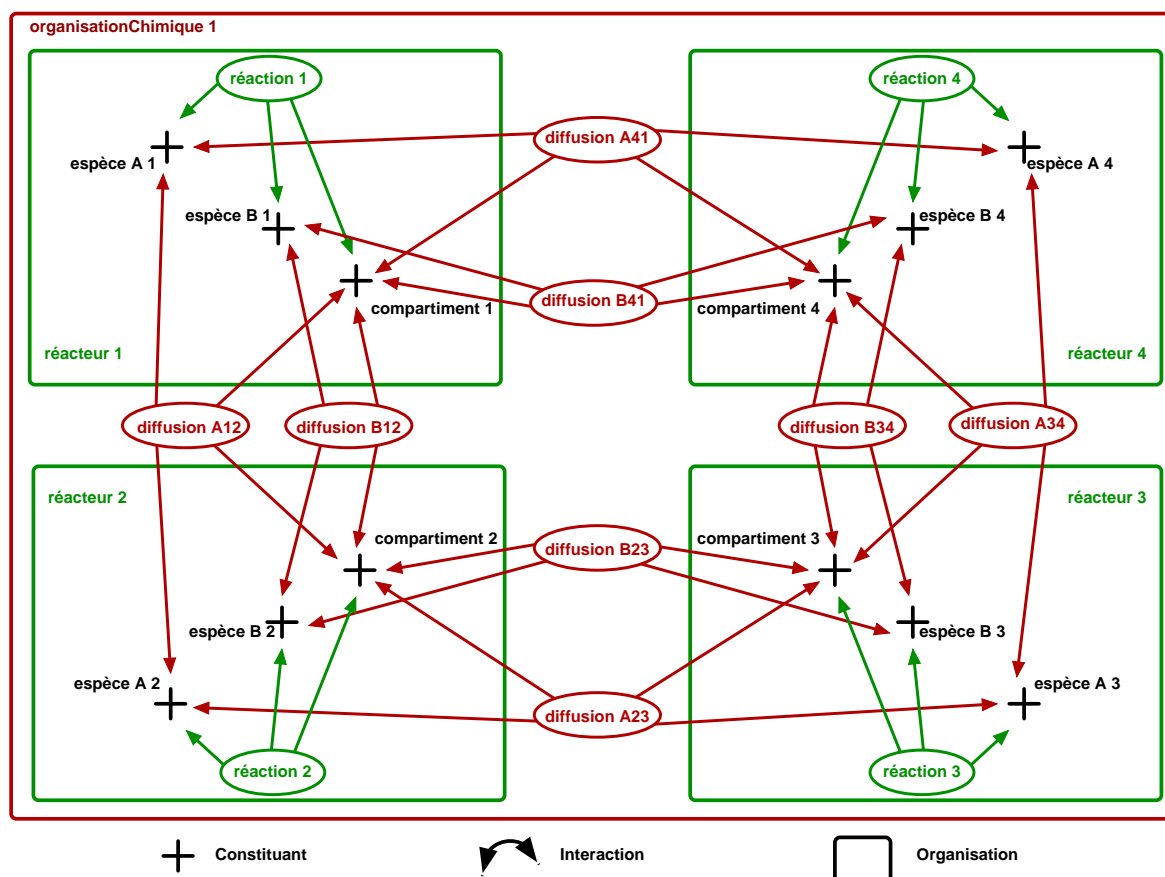


FIG. 5.6: L'organisation `OrganisationChimique` crée des sous-organisations `Réacteur` qui sont le lieu de réactions chimiques. Les interactions `Diffusion` assurent le transport de matière d'un réacteur à l'autre. `Espèce X i` correspond à l'espèce X dans le réacteur i. Les réactions et la diffusion s'opèrent à des niveaux d'organisation différents.

5.1.4 Organisation chimique

L'expérience acquise au cours des différentes applications a donné naissance à un modèle d'organisation particulier : l'« organisation chimique ». Cette organisation sert à définir un milieu chimique discrétisé dans lequel pourront se dérouler les expériences *in virtuo*. Nous considérons le milieu discrétisé comme la juxtaposition de réacteurs chimiques identiques. L'organisation contenant tous ces réacteurs applique le phénomène de la diffusion (voir figure 5.6).

Dans un premier temps, nous considérons des réacteurs associés à des compartiments de forme cubiques². La question de l'adjacence entre compartiments se posant souvent (pour l'organisation cellule notamment), nous avons doté la structure d'un mécanisme permettant de stocker cette information. Ainsi, le phénomène lié à la diffusion peut interroger la structure pour lui demander quels sont les compartiments voisins et quelles sont les espèces associées à chaque compartiment. Enfin, afin de simplifier la modélisation du milieu, chaque réacteur est du type `ModèleSBML` s'appuyant sur le même fichier : `milieu.xml`.

Le milieu chimique étant défini, nous pouvons maintenant le coupler avec des modèles de cellule.

²Dans la partie application, nous proposerons un maillage plus élaboré.

5.2 Organisation de la cellule

5.2.1 La cellule chimique

Le modèle de la cellule se situe à un niveau de modélisation différent de celui des phénomènes chimiques. Pouvoir définir un modèle de cellule dans le même cadre générique qui sert à la modélisation des réactions chimiques nous montre comment il est possible de faire cohabiter plusieurs niveaux d'organisation. Dans un premier temps, nous considérons la cellule comme un système de second ordre composé de sous-systèmes chimiques, ce qui est un bon début puisque c'est souvent la seule nature chimique qui est abordée dans la littérature.

Pour faciliter la définition du modèle, nous nous basons encore une fois sur la norme SBML. Ainsi, l'organisation `Cellule` dérive de la classe `ModèleSBML`. Le fichier SBML associé doit donc définir un réseau biochimique constitué d'un certain nombre de compartiments suivant le modèle souhaité. Ici, nous imposons que ce modèle possède un compartiment nommé « milieu ». Les interactions chimiques de la cellule avec son milieu sont alors naïvement décrites par la norme comme des réactions impliquant des espèces du compartiment « milieu » et d'autres compartiments (par exemple les compartiments « membrane » ou « cytoplasme »). Lors de l'interprétation du fichier SBML, les modèles de réaction seront traduits en phénomènes chargés de créer les objets actifs `Réaction`.

Ensuite, pour que la cellule interagisse effectivement avec son milieu, elle doit maintenir à jour sa topologie. C'est-à-dire qu'elle doit remplacer les constituants `Compartiment` et `Espèce`, créés par défaut lors de l'interprétation du fichier SBML, par les constituants correspondant au milieu dans lequel se trouve la cellule. La cellule est alors en couplage structurel avec le milieu.

La figure 5.7 montre le type de modèle que l'on peut réaliser avec l'organisation `Cellule`. Chaque interaction représente une réaction ou une règle définie avec SBML. Si l'expressivité de la norme ne suffit pas, le modélisateur peut définir ses propres phénomènes sur une base identique à celle des phénomènes de réactions chimiques. Ils créeront alors des interactions agissant selon un algorithme adapté. Par exemple, on peut souhaiter qu'une interaction d'activation soit régie par un simple mécanisme à seuil.

5.2.2 Couplage de la cellule avec le milieu

Nous avons précisé que la cellule devait maintenir sa topologie structurelle en faisant en sorte que le compartiment « milieu » et les espèces associées (tous définis par défaut) devaient être remplacés par ceux correspondant à la position de la cellule. Nous voici donc confrontés à la définition d'un mécanisme de maintien de la topologie par une organisation : la cellule.

Considérons tout d'abord que la cellule évolue dans un univers structuré par une organisation chimique (voir section 5.1.4). Pour déterminer à quel endroit du milieu la cellule se situe, il faut la doter d'une position ; c'est le constituant « position » que l'on peut observer dans la figure 5.7. Ensuite, il est possible de réutiliser le même mécanisme de l'objet `structure` qui a servi au phénomène de la diffusion dans l'organisation chimique. De cette façon, plutôt que de consulter tous les constituants `Compartiment` pour savoir lequel est le plus proche, la cellule interroge la structure. Enfin, un mécanisme d'événement est placé sur le constituant « position ». Chaque fois que la position est perturbée, l'organisation `Cellule` est prévenue et vérifie si ses composants « milieu » et `Espèce` sont toujours adéquats.

Le modèle générique permet de réaliser le changement de milieu extrêmement facilement. Il suffit simplement de retirer les constituants concernés (les différentes interactions concernées

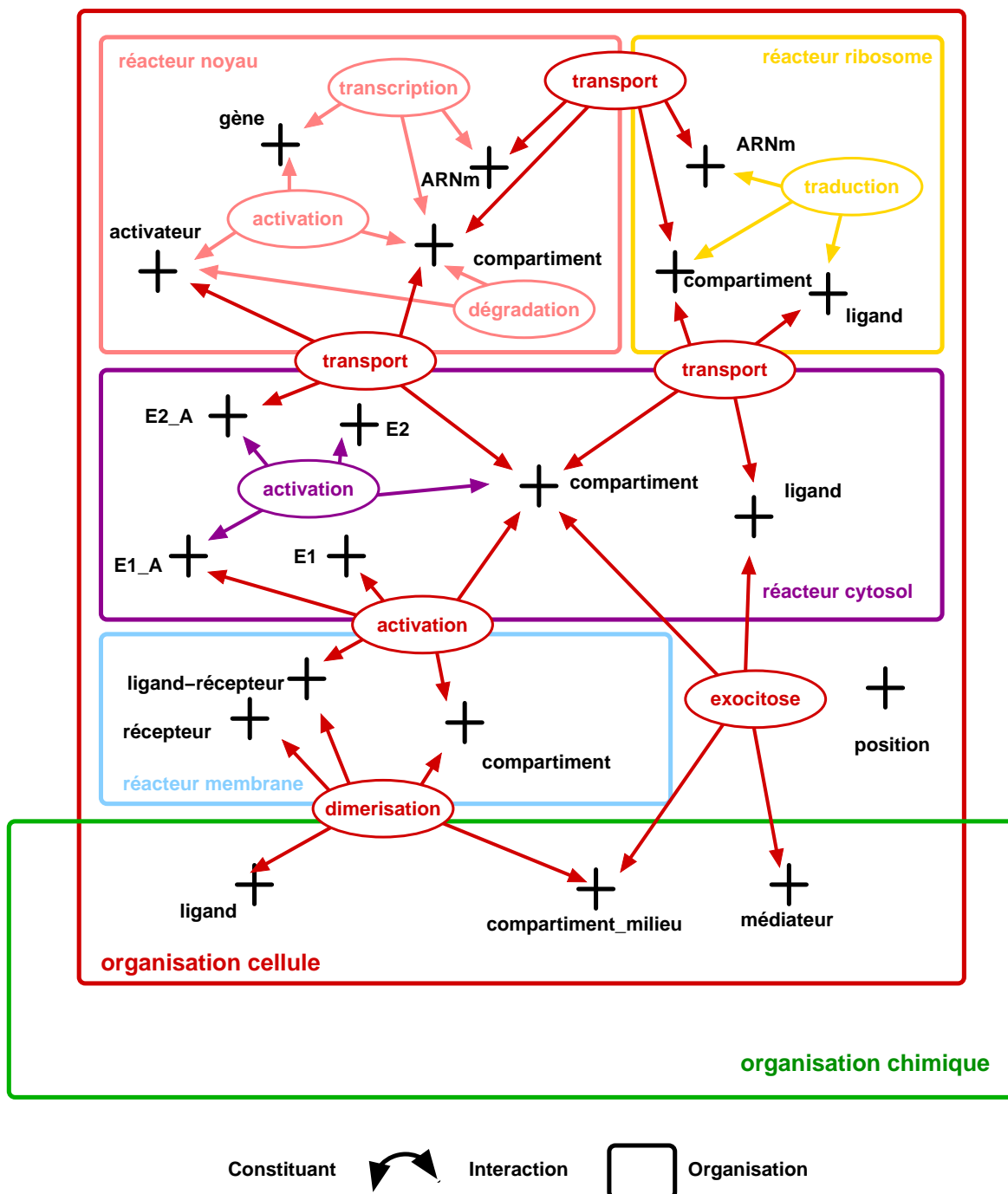


FIG. 5.7: La figure représente le type de modèle que l'on peut définir avec l'organisation *Cellule*. Chaque interaction représente une réaction chimique définie avec la norme SBML. On y distingue une liaison ligand récepteur, une ébauche de cascade enzymatique assurant la transduction dans le cytosol, l'activation d'un gène, la traduction, la transcription et l'exocytose d'un médiateur. On notera la présence d'un constituant « position » qui permet à la cellule de se positionner et d'assurer que les espèces du milieu sont présentes dans sa structure. Ceci étant, ce schéma de principe n'indique en rien que la biologie connaît les lois mathématiques qui régissent ces interactions.

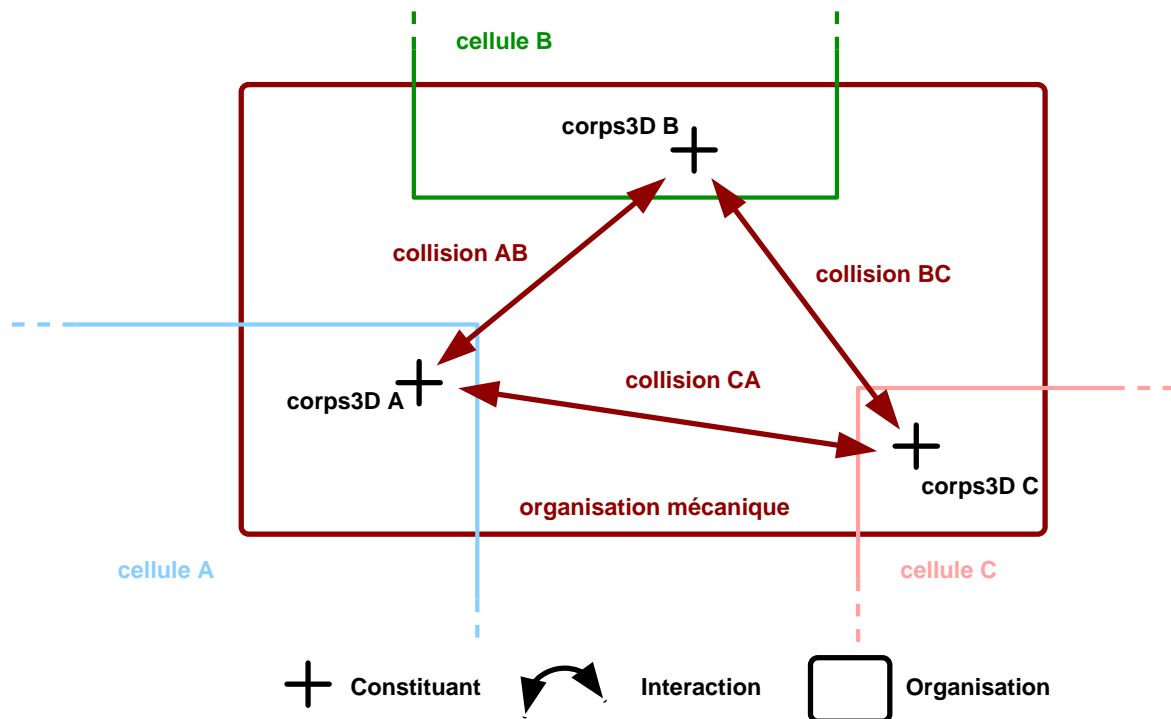


FIG. 5.8: L'organisation mécanique fait interagir les constituants `Corps3D` qui peuvent par exemple appartenir à des organisations `Cellule`.

se « suicideront ») et d'ajouter les nouveaux constituants (les phénomènes se chargeront d'instancier les nouvelles réactions).

Étudions maintenant comment la position de la cellule peut se trouver perturbée.

5.3 Organisation mécanique

5.3.1 Introduction de modèles mécaniques

Plutôt que de doter la cellule d'un simple constituant position, il nous paraît souhaitable de la doter d'une existence physique dans l'univers en trois dimensions. Ainsi, nous définissons un nouveau type de constituant, le `Corps3D`. Un corps est une forme en trois dimensions à laquelle sont associés une position, un repère de référence et une masse. Par l'intermédiaire de leur corps, les cellules peuvent alors interagir dans une organisation mécanique (voir figure 5.8).

La figure 5.9 montre donc l'introduction de quatre nouvelles classes : `Corps3D`, `Collision`, `CollisionPhénomène` et `OrganisationMécanique`.

L'organisation mécanique maintient sa topologie en s'inscrivant auprès de la structure pour recevoir un événement à la création de chaque corps. Le mécanisme garantit alors que tous les corps du monde virtuel font partie de l'organisation.

Deux types de corps ont été créés : sphérique et triangulaire. La forme sphérique est adaptée à la modélisation de cellules. La forme triangulaire permet de construire d'autres formes composées de facettes.

Le phénomène `CollisionPhénomène` fait ses calculs sur les sphères englobantes. Lorsque

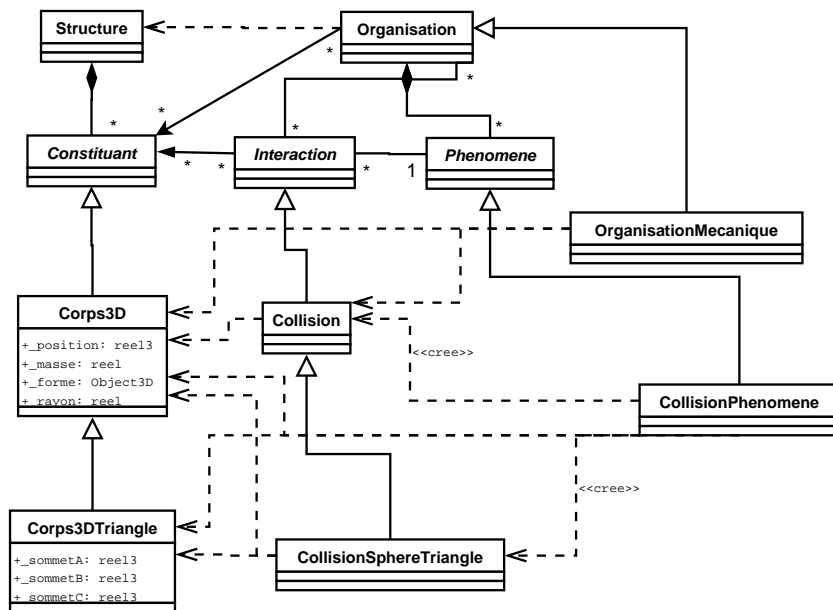


FIG. 5.9: Le diagramme UML représente l'héritage des classes permettant la réalisation des collisions mécaniques.

deux sphères englobantes sont à proximité, le phénomène crée une interaction collision adaptée (collision entre corps indéformables : sphère/sphère ou triangle/sphère...).

Une interaction de collision mécanique a pour but de repousser deux corps si elle perçoit qu'ils sont en contact. Plusieurs solutions sont alors envisageables pour repousser les corps. Pour la collision entre sphères nous avons choisi un déplacement élastique dépendant de la distance de contact, de la différence de masse et du temps. De cette façon nous pouvons par exemple observer l'évolution d'amas de cellules (voit figure 5.10).

Avec le même principe, il est possible d'établir des interactions d'adhésion (entre corps), ou d'agitation (entre un corps et un compartiment « milieu »).

5.3.2 Optimisations

La détection des collisions est toujours un problème extrêmement coûteux lors de la simulation de modèles en trois dimensions. Pour être capable d'introduire un nombre conséquent de corps dans un univers réalisant déjà la simulation d'autres phénomènes (chimiques par exemple), il faut utiliser un certain nombre de techniques d'optimisation des calculs de détection des collisions.

La première optimisation, celle que nous avons précédemment indiquée, est le fait que le phénomène reçoit un événement à chaque ajout/suppression d'un corps. Ainsi le phénomène détient la liste de tous les corps présents.

Le phénomène peut alors structurer ces corps dans une grille en trois dimensions. Le maillage de la grille possède un pas supérieur au plus grand diamètre de la sphère englobante du plus grand des corps. À chaque mouvement d'un corps un événement est émis. Il est reçu par le phénomène qui vérifie si le corps est toujours dans la même case. S'il change de case, alors il crée les nouvelles interactions *Collision* nécessaires.

Ensuite, l'activité de l'interaction *Collision* n'a pas forcément besoin d'avoir une fréquen-

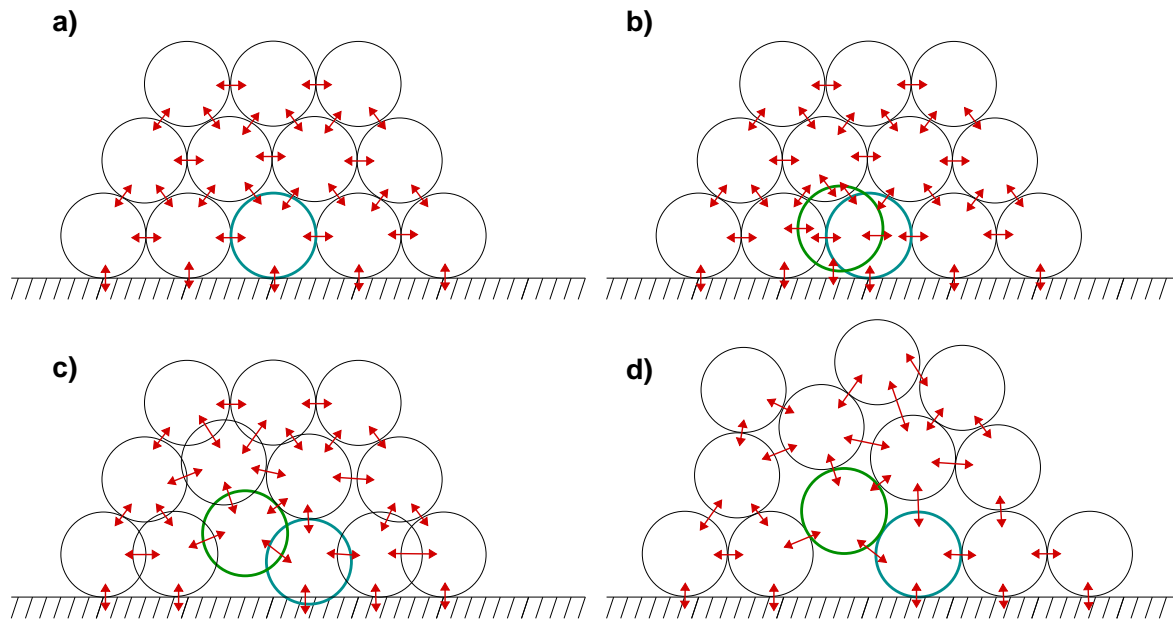


FIG. 5.10: La figure représente un exemple d'application que permet le système de gestion des collisions. On observe ce qu'il se passe lorsque la cellule bleue se divise en une cellule bleue et une verte.

ce d'activation aussi élevée que les réactions chimiques. En effet, lorsqu'on souhaite modéliser des systèmes tissulaires au niveau cellulaire, il y a finalement assez peu de mouvement. Là où les réactions chimiques se réalisent à des pas de temps de l'ordre de la milliseconde, les collisions peuvent intervenir tous les dixièmes de seconde voire toutes les demi secondes.

Enfin, dans le cas où la collision agit sur deux objets très proches mais qui demeurent immobiles, il n'est pas nécessaire de vérifier leur position en permanence. L'interaction peut alors s'inscrire au même événement que précédemment (qui se déclenche lorsqu'un corps est déplacé) et s'endormir pour ne plus consommer de ressources de calcul jusqu'à ce que l'un des corps qui la concerne ne la réveille en bougeant. Ce mécanisme allège fortement la simulation quand il y a très peu de mouvement.

Le choix des méthodes utilisées a été effectué en fonction des applications réalisées. C'est un compromis entre temps de calcul et développement. À chaque application, il faut faire un compromis pour trouver la solution et les paramètres optimaux. Ce qu'il faut retenir, c'est que toutes les spécificités inhérentes à la détection des collisions sont encapsulées dans la classe `CollisionPhénomène` et la classe `Collision`. Si le modélisateur souhaite modifier ce phénomène, alors il peut le faire de manière indépendante ; c'est l'intérêt de la modularité et de la réification des interactions.

5.3.3 Interaction ponctuelle, cas limite d'utilisation

La prise en compte des collisions est intéressante parce qu'elle s'attaque à un cas limite d'application du modèle proposé. Il existe de nombreux travaux sur la détection des collisions [Meseure and Kheddar, 2006]. En général, le traitement des collisions se fait lors du déplacement des entités ; on détecte si la position d'arrivée est valide. S'il elle ne l'est pas, on traite la collision en conséquence. La collision est donc instantanée.

Le fait de réifier les interactions n'est pas adapté aux interactions ponctuelles. Il est évidemment inutile de créer une instance d'interaction si elle n'agit qu'une seule fois sur le

système, autant réaliser directement l'interaction. Deux solutions s'offrent alors à nous pour traiter le phénomène événementiel de la détection des collisions : chercher à l'adapter à notre modèle ou lui adapter notre modèle.

Dans un premier temps, nous nous sommes donc interrogés sur ce que devenait le phénomène de collision dans le modèle générique. Le principe d'autonomie nous amène à traiter séparément le phénomène de collision et d'autres phénomènes éventuels qui déplaceraient les corps. Il ne s'agit alors plus de traiter la collision pendant le déplacement, puisque ce sont deux actions qui sont indépendantes, mais de créer une interaction là où il peut y avoir un recouvrement. C'est ce que nous venons de présenter. L'avantage de cette façon de faire est de pouvoir observer simplement le résultats d'un nombre conséquent de collisions à partir d'interactions élémentaires, là où une gestion classique aurait des difficultés à proposer un modèle physique réaliste ou alors risquerait d'être redondante (A interagit avec B et B interagit avec A). L'application décrite au chapitre 6 montrera de plus que cette gestion des collisions est largement suffisante en terme de performance pour nos besoins.

La solution implémentée contourne alors le problème des interactions ponctuelles en les rendant persistantes et dépendantes du temps. Toutefois, il peut être bon d'imaginer une solution pour modéliser ce genre d'interaction pour le cas où l'on ne pourrait contourner le problème. La solution apparaît d'elle-même si l'on revient à la définition de l'interaction que nous avons donnée au chapitre précédent. L'interaction est en effet perçue comme une manifestation particulière d'un phénomène. Pour éviter que le phénomène ne perde du temps à instancier une interaction à activation unique, il est possible d'imaginer que le phénomène réalise directement cette action. Ainsi, il y aurait bien une interaction ponctuelle. Cela signifie que l'on permet au phénomène de modifier directement la structure du modèle.

La raison pour laquelle nous n'avons pas encore introduit cette possibilité dans le modèle générique est que nous n'avons décrit que l'ordonnancement des interactions. Cela mériterait donc une réflexion sur l'ordonnancement des actions réalisées par les phénomènes sur les constituants. Autrement dit, il faudrait s'assurer que l'action d'un phénomène soit strictement équivalente en terme de résultat, à la création d'une interaction, à son activation par l'ordonnanceur, puis à son suicide. De plus, réaliser concrètement cet aspect du modèle n'est pas une priorité actuellement. Nous préférons donc ne pas le formaliser dans cette thèse qui n'est qu'une étape, nous gardons néanmoins cette possibilité en perspective. Il temps maintenant de nous pencher sur « le » modèle de la réalité virtuelle : l'utilisateur.

5.4 L'humain dans la boucle

L'objectif premier de ce travail est l'établissement d'un modèle générique de modélisation des systèmes vivants pour la réalité virtuelle. La prise en compte de l'utilisateur dès la construction du modèle est fondamentale pour assurer la dimension réalité virtuelle. Dans cette section, nous montrons en quoi la prise en compte de l'utilisateur est immédiate. Nous montrons ensuite comment l'utilisateur/modélisateur peut agir sur le modèle. Enfin, nous élargissons à la notion d'interface.

5.4.1 L'utilisateur : médiation des sens et de l'action

L'utilisation du principe d'autonomie pour la construction des modèles en réalité virtuelle nous conduit à modéliser l'utilisateur. En général, cet acte se traduit par la création d'un

avatar de l'utilisateur. Ensuite, tout ce passe comme s'il était un modèle autonome parmi les autres.

Nous avons construit nos modèles comme un ensemble de systèmes autonomes en couplage structurel. Cette façon d'appréhender la modélisation inspirée des travaux de Varela sur la modélisation du vivant, possède l'avantage de très bien s'adapter à l'idée d'énaction présentée dans l'état de l'art (voir section 3.1.1). Ainsi, pour permettre à l'utilisateur de faire l'expérience du modèle numérique, il faut le mettre en couplage avec ce dernier.

Dans le cadre du modèle générique que nous nous sommes fixés, le couplage entre l'utilisateur et le modèle est très simple. Il suffit de lui donner accès aux objets **Constituants**. Chaque modification d'état que pourra exécuter l'utilisateur sera perçue comme une perturbation par les autres systèmes autonomes. Il apparaît que la notion d'avatar n'est plus utile dans ce cas, puisqu'aucune entité du modèle n'a besoin de connaître explicitement l'utilisateur.

Finalement, afin d'augmenter au maximum le champ d'expérience de l'utilisateur, nous lui donnons accès par défaut à l'ensemble de la structure de l'univers virtuel. La médiation des sens est donc réalisée par la mise à disposition de l'utilisateur de l'information indiquant l'état des constituants. La médiation de l'action est elle, permise par la possibilité offerte à l'utilisateur de modifier ou de créer ces états, c'est le rôle de l'interface.

5.4.2 Le modélisateur : médiation de l'esprit

La troisième médiation à réaliser pour appliquer pleinement les possibilités offertes par la réalité virtuelle est la médiation de l'esprit. C'est-à-dire que l'utilisateur doit aussi pouvoir être modélisateur. La solution idéale pour offrir cette possibilité est de permettre le prototypage interactif. C'est d'ailleurs ce que proposaient les travaux préliminaires à cette thèse par l'utilisation du langage oRis [Harrouet et al., 2002; Desmeulles, 2003]. Deux raisons nous ont poussés à abandonner cette possibilité : d'abord le prototypage interactif nécessite l'usage d'un langage interprété ce qui augmente considérablement les besoins en puissance de calcul ; ensuite, nous souhaitons que l'utilisateur/modélisateur soit un biologiste, il n'est pas souhaitable qu'il modifie le modèle informatique mais simplement le modèle biologique. Ceci implique qu'il n'y a pas à créer de nouvelle classe pendant la simulation, l'utilisateur/modélisateur doit simplement composer avec les instances des classes préexistantes.

Nous voyons donc apparaître deux niveaux de modélisation : la modélisation informatique et la modélisation biologique. L'utilisateur ne doit avoir accès qu'à la deuxième. C'est à dire qu'il peut intervenir sur les instances des organisations, des phénomènes, des interactions, des constituants et de la structure. La structure et les constituants sont le domaine d'action que nous avons défini pour la médiation des sens et de l'action. Il nous reste donc à aborder les trois autres points : organisations, phénomènes et interactions.

Tout d'abord, les organisations sont les principaux objets d'action du modélisateur. L'objet **Organisation** possède un certain nombre de fonctions permettant à tout moment de lui ajouter ou enlever des phénomènes, des interactions, des sous-organisations. Il est également possible de modifier sa structure en modifiant l'ensemble des objets **Constituant** qu'elle connaît. Pour chacune des actions citées, des mécanismes de mise à jour ont été développés pour permettre la prise en compte des actions de modélisation pendant l'exécution des simulations.

Ensuite, il est possible de modifier les objets **Phénomène** qui définissent les conditions d'apparition des interactions. Dès lors, il est possible d'inhiber ou de favoriser une interaction en jouant sur les paramètres du phénomène associé. Cette possibilité est très intéressante

pour comprendre le rôle d'une interaction dans un système complexe.

Enfin, il est possible de vouloir jouer sur certains paramètres de certaines instances d'interaction. Ceci ne comporte pas de limite technique mais ne correspond pas à la philosophie préconisée. En effet, les interactions doivent émerger des phénomènes. Si on veut modifier leurs vitesses, paramètres... cette action doit se réaliser au niveau du phénomène qui instancie les interactions. Il reste cependant possible d'agir sur une instance particulière d'interaction.

Les interactions `Réaction` fournissent un bon exemple de ce que l'on peut utiliser pour modifier les interactions (voir la section 5.1.4). En effet, le même mécanisme qui permet de créer l'organisation `ModèleSBML`, les phénomènes `RéactionPhénomène` et les interactions `Réaction`, peut être utilisé pour redéfinir certaines parties du modèle. Ajoutons que lorsqu'on modifie un phénomène, il est en charge de demander à toutes les interactions concernées de se suicider pour pouvoir en générer de nouvelles.

Nous voyons paraître ici un avantage de notre choix de modélisation par rapport aux modèles agents basés entités et non interactions. En effet, dans un système multi-agents classique les phénomènes physiques ne sont réalisés que si chaque entité possède dans son comportement les algorithmes permettant leur mise en œuvre. La notion de phénomène global n'existe pas, ce n'est qu'une abstraction que nous déduisons en généralisant le comportement des entités. Modifier, ajouter ou supprimer une loi dans l'univers virtuel revient alors à modifier l'ensemble des entités qui le peuple ou à imposer un contrôle global. Dans notre cas, nous modélisons directement cette loi qui s'intègre de manière autonome dans le modèle car c'est l'influence de cette loi que nous voulons expérimenter. Il est appréciable de pouvoir manipuler directement le phénomène plutôt qu'une multitude d'objets, surtout pour permettre la médiation de l'esprit en temps réel.

Finalement, les principes d'autonomie et de réification des interactions nous permettent de réaliser à peu près toutes les actions de perturbation et de modification du modèle pendant son exécution, à condition que ces modifications entrent dans le champ des possibilités des classes déjà définies. Il appartient à l'informaticien qui crée le modèle numérique de se poser la question de la liberté de modélisation qu'il souhaite laisser au biologiste expérimentateur. Ensuite, il suffit de posséder les interfaces adéquates.

5.4.3 À propos des interfaces

Le rôle des interfaces est donc de permettre la médiation des sens, de l'action et de l'esprit. L'établissement du modèle générique que tend à exposer cette thèse est une des étapes qui mène à la réalisation d'un laboratoire virtuel. Nous n'avons pas étudié précisément les interfaces dans ce travail. La réflexion sur la conception d'interfaces élaborées pour l'expérimentation *in virtuo* mérite bien de faire l'objet d'une thèse à part entière. De plus, la réalisation d'interfaces est extrêmement liée aux cas d'application. Or, nous essayons de nous placer dans le cas général. Cependant, certaines pistes ont été ouvertes pour la mise en œuvre des interfaces ; celles-ci ayant valeur de test. Ainsi, nous vérifions que le modèle permet la médiation des sens, de l'action et de l'esprit.

A titre d'exemple, nous pouvons citer la fenêtre de visualisation des modèles 3D, fournie par la bibliothèque ARéVi. Elle permet à l'utilisateur d'observer l'évolution des corps des différentes espèces cellulaires. Certaines informations peuvent être ajoutées : visualisation du maillage du milieu ; modification de la couleur en fonction du niveau d'activation... Il est également possible d'afficher sur une courbe les concentrations chimiques à l'endroit où l'utilisateur positionne une « sonde ». Enfin, nous avons modélisé une seringue virtuelle qui permet à l'utilisateur d'ajouter une concentration moléculaire à l'endroit où il le désire,

n'importe quand dans les simulations. La partie application de la thèse illustrera ces quelques interfaces.

5.5 La libBio

Le modèle générique ainsi que les différents modèles présentés dans ce chapitre ont été implémentés dans une bibliothèque que l'on nommera de manière très originale : `libBio`. La `libBio` est donc une bibliothèque C++ utilisant elle-même la bibliothèque ARéVi. Se faisant, elle peut réutiliser tout l'arsenal d'outils dédiés à la réalité virtuelle (et éprouvés) que fournit la plate-forme ARéVi (ordonnanceur, Lib3D, gestion des événements...). La `libBio` contient donc en premier lieu les classes mères définissant le cœur du modèle : `Interaction`, `Organization`, `Phénomène`, `Constituent` et `Structure`. Nous passons là du modèle théorique au modèle implémenté ce qui ne va jamais sans quelques modifications du modèle initial. Cependant, nous pensons que ce dernier respecte très bien le modèle théorique que nous venons de décrire, et nous espérons que la transgression la plus importante que nous ayons concédée soit le passage des noms de classe en langue anglaise³. La figure B.1 présentée dans les annexes fournit un diagramme de classes détaillé des classes de base de la bibliothèque.

Ensuite, nous intégrons dans la `libBio` au fur et à mesure des applications, les modèles qui présentent un aspect réutilisable ; ceci pour constituer un ensemble d'outils logiciels pour l'expérimentation *in virtuo*. Les différents modèles présentés dans ce chapitre composent donc cette boîte à outils. La figure 5.11 résume le contenu de la bibliothèque. Les figures B.2 et les figures B.3, données dans les annexes, présentent en détail les classes contenues dans la `libBio`.

Le chapitre précédent nous a donné un modèle générique, ce chapitre nous a donné des outils de bases. Ces outils sont de nature chimique : réactions, diffusions, compartiments... et de nature mécanique : collisions, corps... Il nous reste maintenant à utiliser la `libBio` pour construire les applications permettant l'expérience *in virtuo* de phénomènes biologiques. Ainsi nous montrerons par sa mise en œuvre que le cadre générique est applicable.

³Puisque nous nous référons dorénavant à la `libBio`, les noms de classes dans les diagrammes UML seront donnés en anglais car ce sont les noms effectivement implémentés.

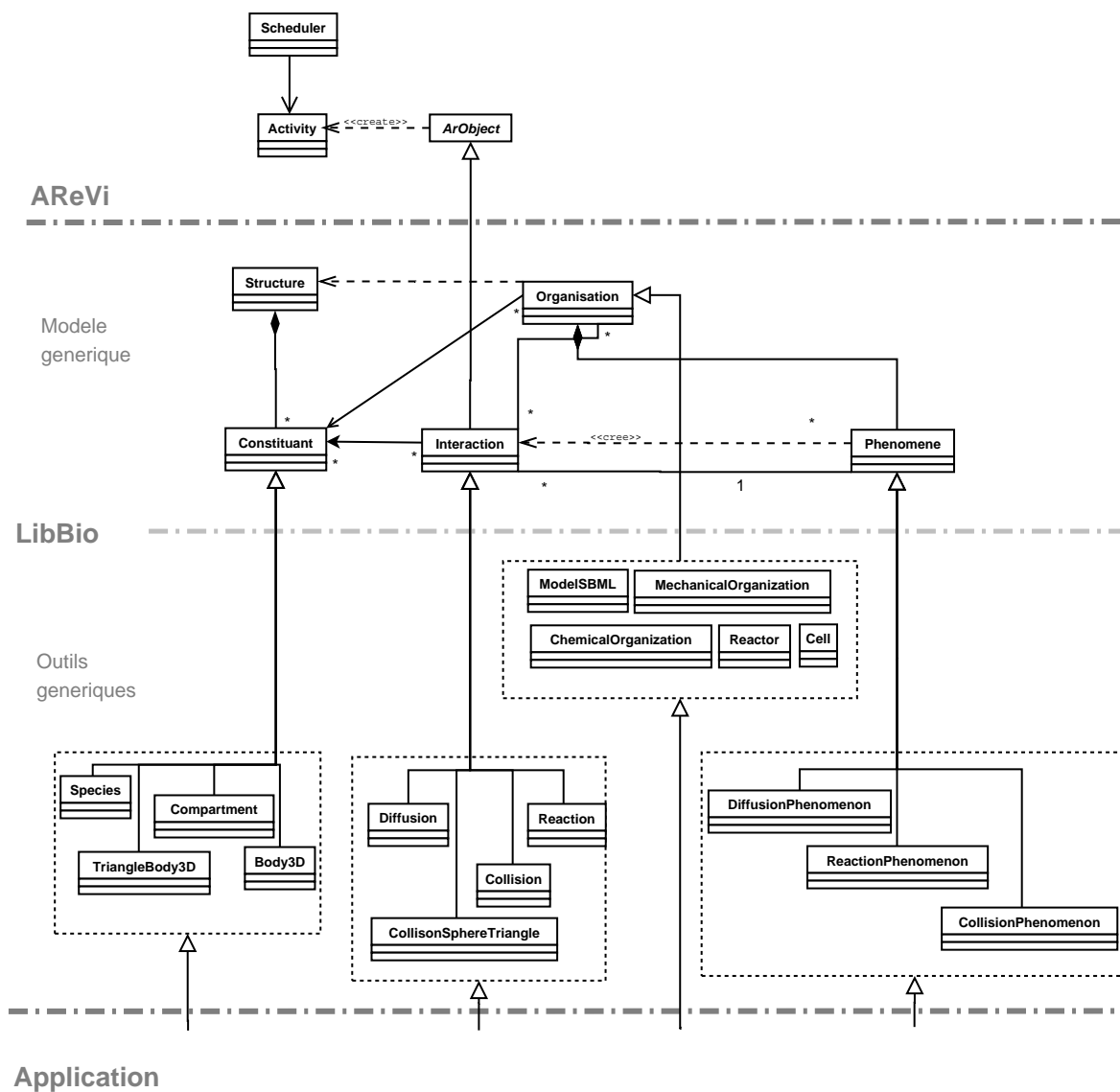


FIG. 5.11: Architecture de la libBio.

Troisième partie

Applications

Chapitre 6

Derme in virtuo

Cette thèse s'inscrit dans le contexte d'une collaboration entre l'équipe Écosystémique et Biologie Virtuelles (EBV) du Centre Européen de Réalité Virtuelle (CERV) et l'équipe de neurobiologie cutanée appartenant à l'Unité de Physiologie Comparée et Intégrative (EA3879) du Centre Hospitalier Universitaire de Brest. Cette dernière équipe, dirigée par le professeur Laurent Misery a pour sujet d'étude les interactions entre la peau, le système immunitaire et le système nerveux.

6.1 Contexte biologique.

Le système immunitaire regroupe différents types cellulaires, dont les actions coordonnées permettent notamment au corps humain de lutter contre les agressions extérieures, qu'elles soient virales ou bactériennes. L'élimination de ces agents infectieux s'accompagne de phénomènes comme l'inflammation, ou le prurit. Dans le derme humain, ce système est représenté par des cellules effectrices, comme les macrophages, qui par la phagocytose permettent l'élimination bactérienne, ou les lymphocytes T qui réduisent le risque viral et bactérien. Il existe aussi des cellules signalisatrices, comme les mastocytes ou les éosinophiles, qui stimulées par la présence d'agents infectieux, libèrent dans le milieu de nombreuses substances influant sur l'activation de cellules effectrices voisines.

Le système nerveux est lui responsable de la transmission des informations nerveuses, et sa présence dans la peau se fait sous forme de différents types de terminaisons qui permettent de faire circuler des informations entre la périphérie (muscle, peau...) et le système nerveux central (moëlle épinière et cerveau), dans un sens comme dans l'autre, selon le type de la fibre. Ces fibres nerveuses sont sous l'influence, par l'intermédiaire de récepteurs présents sur la membrane, de différentes molécules que l'on peut retrouver dans le derme. L'un des effets de cette stimulation des fibres nerveuses est un relargage de neuropeptides, peptides pouvant interagir avec différents types cellulaires, présents à proximité de la terminaison de la fibre.

Ensuite, la peau est un organe composé de plusieurs couches de tissus. Elle joue, entre autres, le rôle d'enveloppe protectrice du corps. Chez l'homme, elle est l'un des organes les plus importants du corps en regard de sa surface et de sa masse : chez l'adulte, environ $2m^2$ pour 5kg. Sur le plan anatomique, la peau comprend deux parties principales. La partie superficielle mince s'appelle l'épiderme, rattachée à une partie interne plus épaisse : le derme. Une troisième couche, plus profonde, constitue l'hypoderme mais n'est classiquement pas assimilée à une couche de peau. La peau remplit de nombreux rôles : régulation de la

température, protection de l'organisme, perception (chaleur, pression, douleur, prurit).

Ce découpage en différents systèmes (immunitaire, nerveux, cutané) est très ancien, mais aujourd'hui, la limite n'apparaît plus aussi nette tant ces systèmes interagissent. Ces interactions sont fortes puisqu'il existe d'une part des connexions anatomiques entre les cellules des 3 systèmes, et d'autre part des connexions fonctionnelles, puisque les cellules de ces 3 systèmes possèdent des récepteurs leur permettant de capter des messages biochimiques provenant des autres systèmes. Ces messages sont portés par des neuromédiateurs, des neurotrophines et des cytokines. Ces interactions sont observables dans les circonstances normales et au cours de maladies.

Un bon exemple en est l'allergie, en particulier l'urticaire. L'urticaire est une réponse de la peau à une substance étrangère, qu'elle reconnaît comme étrangère. Le caractère étranger de l'allergène est reconnu grâce à des cellules immunitaires *spécifiques* : les lymphocytes B. Puis interviennent d'autres cellules immunitaires : macrophages, éosinophiles, etc... Un médiateur important des interactions cellulaires en cas d'urticaire est l'histamine. L'histamine est principalement libérée par les mastocytes, sous l'action de substances libérées par les terminaisons nerveuses dans la peau. L'histamine augmente nettement les mécanismes mis en jeu dans l'urticaire, ou même les initie : vasodilatation, œdème, afflux de cellules.

6.2 Intérêts mutuels du projet

Nous avons choisi d'établir un modèle du phénomène de l'urticaire allergique comme support la mise en œuvre des connaissances biologiques de l'équipe de neurobiologie cutanée. L'application ainsi réalisée présente nombre d'intérêts pour les deux parties de la collaboration entre biologistes et informaticiens.

D'un point de vue biologique, ce sont les possibilités d'intégration des connaissances qui présentent le premier intérêt. En effet, il s'agit d'intégrer dans un même modèle des connaissances issues de sources bibliographiques variées ou recueillies auprès de différents chercheurs. Ainsi l'application permet à long terme de faire le lien entre les différents volets des travaux de l'équipe de recherche. Elle fournit un support au travail interdisciplinaire qui doit permettre dans l'avenir, de proposer de nouvelles hypothèses.

D'un point de vue informatique, la réalisation de l'application guide la partie plus générique de nos travaux, car finalement, la seule façon fiable de tester et de mettre au point une méthode de modélisation, c'est sa mise en pratique. Nous pourrons dès lors répondre aux questions : le modèle générique est-il suffisamment expressif ? Sa mise en œuvre est-elle ergonomique ?

À partir du modèle générique que nous avons proposé, la construction de la version du modèle biologique numérique présentée dans cette thèse relève en grande partie du travail de Nicolas Froment (Biologiste) [Froment, 2005]. Les recherches de données bibliographiques ont constitué une partie importante dans la construction de ce modèle. Afin de produire le modèle le plus complet, nous avons dû multiplier les axes de recherches. Ainsi, les données biologiques ont pu être obtenues aussi bien par contact avec des chercheurs spécialisés dans des domaines intervenant dans le modèle comme F. Guerrero (Maître de Conférences au Laboratoire des Sciences du Sport et des Activités Nautiques de l'Université de Bretagne Occidentale), que par les moyens classiques de recherches bibliographiques (bibliothèques universitaires, PubMed ou BioCarta ...). Nous devons bien sûr citer les différents membres du Laboratoire de Neurobiologie Cutanée qui ont participé à l'établissement d'un modèle :

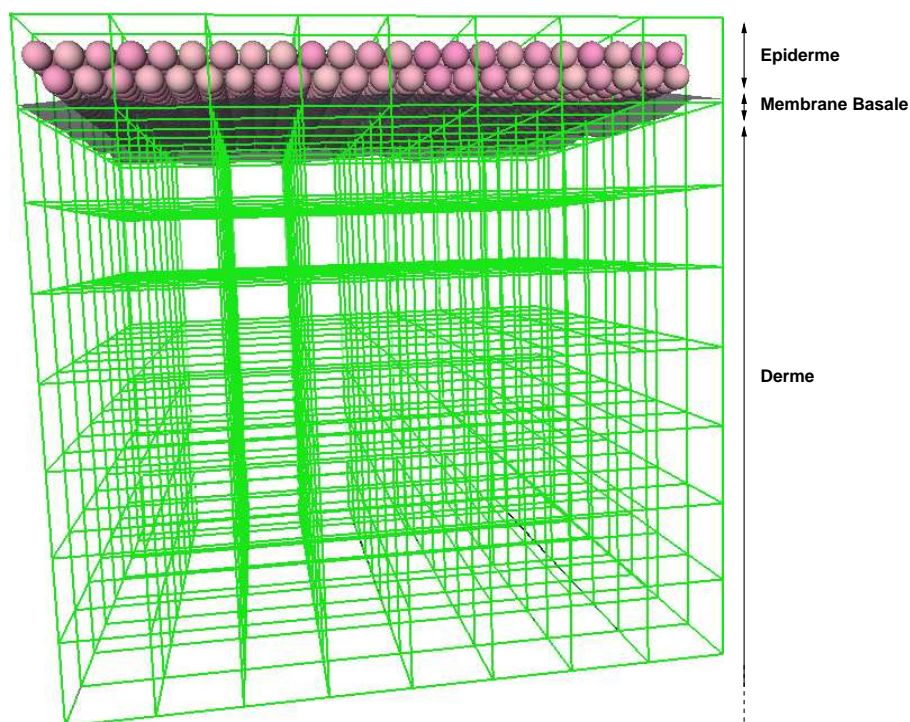


FIG. 6.1: La figure est une visualisation de l'environnement dans lequel se réaliseront les phénomènes étudiés.

Laurent Misery, Anne-Marie Roguedas, Yannick Chateau, Ulysse Pereira, Nicholas Boulais, Dominique Dupré.

Ce manuscrit étant un travail d'informatique, nous ne justifierons pas ici les choix inhérents à la biologie. Nous nous contenterons d'un exposé descriptif montrant comment se construit le modèle dans le cadre proposé par cette thèse.

6.3 L'organisation Derme

Nous avons choisi de modéliser un cube de peau d'un millimètre de côté. Nous avons également choisi de représenter le derme superficiel car il est le lieu de la formation d'un œdème, et le début de l'épiderme. La figure 6.1 montre l'environnement dans lequel nous pourrions faire l'expérience des phénomènes modélisés. À cette étape, notre modèle est une organisation **Derme** composée de quatre parties : l'organisation mécanique, le milieu, les couches de kératinocytes de l'épiderme et la membrane basale. La première de ces organisations est décrite dans la section 5.3. Les trois autres sont décrites brièvement dans cette section.

6.3.1 Le milieu

Le milieu représente le derme. Le derme est un tissu constitué en majorité de collagène que nous ne représentons pas dans ce modèle. Le milieu est alors modélisé par une organisation chimique (voir section 5.1.4). Le fichier SBML associé à cette organisation chimique décrit l'existence des différents médiateurs qui peuvent être présents dans le milieu, ainsi que les réactions de dégradation de ces médiateurs. L'exemple ci-après montre comment on définit un milieu avec un seul type de molécule, l'histamine. La réaction de dégradation permet d'assurer

la demi-vie de l'espèce chimique. Bien sûr, il est possible d'utiliser un éditeur SBML pour générer cette description.

```
<listOfCompartments>
  <compartment id="milieu" size="1"/>
</listOfCompartments>

<listOfSpecies>
  <species id="Histamine_milieu"
    initialConcentration="0"
    compartment="milieu"
    boundaryCondition="false"
  />
</listOfSpecies>

<listOfReactions>
  <reaction id="Demie_Vie_Histamine" reversible="false">
    <listOfReactants>
      <speciesReference species="Histamine_milieu"
        compartment="milieu"
        stoichiometry="1"
      />
    </listOfReactants>
    <kineticLaw>
      <math xmlns="http://www.w3.org/1998/Math/MathML">
        <apply>
          <times/>
          <ci>Histamine_milieu</ci>
          <ci>ka</ci>
        </apply>
      </math>
      <listOfParameters>
        <parameter id="ka" value="0.000385"/>
      </listOfParameters>
    </kineticLaw>
  </reaction>
</listOfReactions>
```

L'organisation chimique interprète le fichier `milieu.xml` et génère plusieurs réacteurs chimiques composant le milieu. C'est le maillage que l'on observe sur la figure 6.1.

6.3.2 L'épiderme

L'épiderme est la partie supérieure du derme. Il se compose essentiellement du type cellulaire que l'on nomme kératinocyte. Les kératinocytes se disposent en couches, de la matrice fixée sur la membrane basale, à la couche supérieure composée de cellules mortes désquamées. Classiquement, la peau humaine comporte six ou sept couches de kératinocytes.

Les évolutions du modèle conduiront certainement à définir une organisation **Epiderme** qui possédera sa propre organisation chimique. Actuellement, seules quelques cellules passives représentant les deux premières couches de kératinocytes sont insérées dans le modèle (voir figure 6.2). La classe **Keratinocyte** dérive alors de **Cell**. Son seul constituant est un **Body3D** sphérique, qui va rendre la cellule sensible aux collisions (voir figure 6.3).

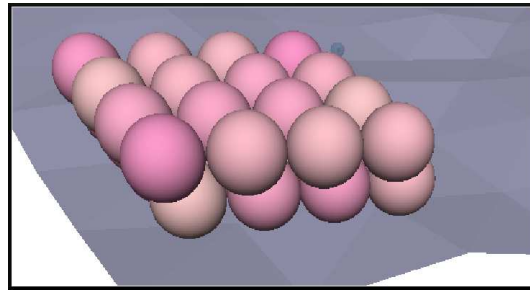


FIG. 6.2: Visualisation des corps des kératinocytes.

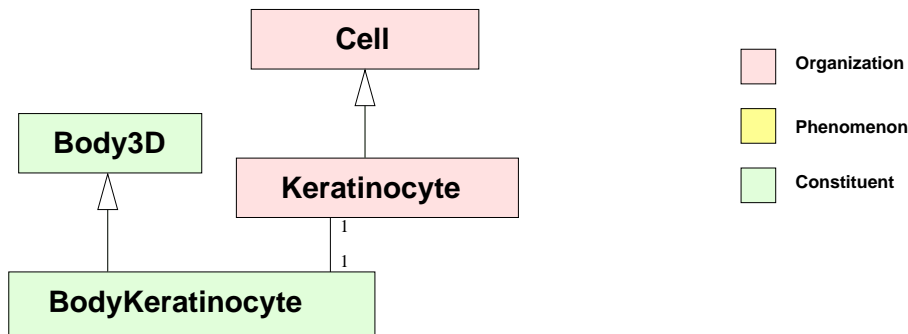


FIG. 6.3: Diagramme de classes des kératinocytes.

6.3.3 La membrane basale

Au niveau de la peau, la membrane basale sépare les cellules épidermiques et le derme. Elle est constituée de molécules telles la laminine, le collagène de type IV et VII, ainsi que des glycosaminoglycanes et des fibres de réticuline en réseau. Sa perméabilité régule les échanges de molécules, en particulier de nutriments, entre les deux tissus. N'ayant pas modélisé le milieu de l'épiderme, cette dernière fonction n'a pas été modélisée. Elle remplit alors seulement son rôle mécanique. Finalement l'organisation `BasalMembrane` est composée d'un ensemble de corps triangulaires immobiles qui empêche les autres corps du modèle de passer d'un tissu à l'autre (voir figure 6.4). Les corps `BodyBasalMembranePart` influent alors sur le modèle dans l'organisation mécanique.

L'intérêt de modéliser la membrane comme un ensemble de triangles, est de pouvoir moduler la forme de la membrane. De plus, grâce au modèle générique, il est possible de changer cette forme à tout moment indépendamment du reste de l'application ; ce qui peut

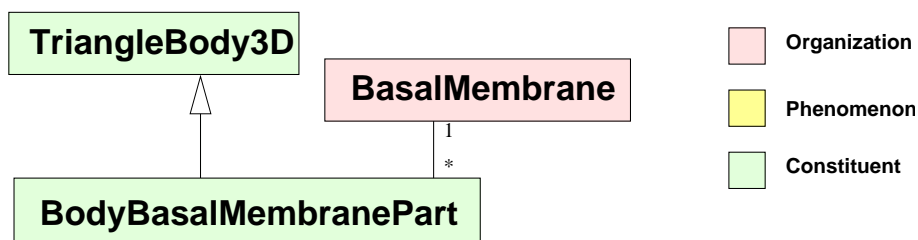


FIG. 6.4: Diagramme de classes de la membrane basale.

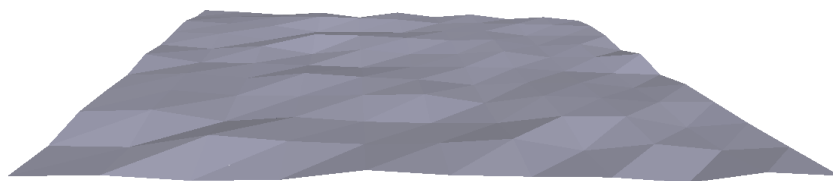


FIG. 6.5: La membrane basale est constituée d'un ensemble de corps triangulaires répartis aléatoirement autour d'un plan.

laisser entrevoir la possibilité de modéliser les phénomènes de déformation de la membrane¹. Ici, les positions initiales des triangles varient aléatoirement autour d'un plan (voir figure 6.5). Nous pourrions² au besoin modéliser les papilles formées par les ondulations de l'épithélium.

6.4 Les mastocytes

6.4.1 Description

Les mastocytes ont un rôle central dans la réaction allergique. Ce sont eux qui produisent et libèrent l'histamine qui est « la » molécule de l'allergie. Leurs modèles numériques occupent donc une place importante dans l'application.

Les mastocytes sont des cellules de forme arrondie, et d'un diamètre d'environ $10\ \mu\text{m}$ [Vervloet and Magnan, 2003]. Ils possèdent une longue durée de vie et une faible capacité mitotique. Ce type cellulaire peut être divisé en deux sous-classes : les MCT qui produisent de la tryptase, et les MCTC (qui eux synthétisent tryptase et chymase). Dans la peau, on retrouve principalement des mastocytes de sous type MCTC [Vervloet and Magnan, 2003]. La partie du derme la plus riche en mastocytes est la zone la plus superficielle du derme (celle que nous modélisons), à la limite avec l'épiderme (épiderme où on ne trouve pas de mastocytes en conditions normales), et plus précisément à proximité des vaisseaux sanguins et des fibres nerveuses [Church and Clough, 1999].

La densité de mastocytes dans la peau varie selon l'âge, le sexe ou la région de la peau. Les chiffres présentés dans la littérature sont très variables : on trouve des concentrations allant de $7 * 10^3$ mastocytes par mm^3 à 46 cellules par mm^3 [Church and Clough, 1999]. D'autres sources parlent de 40 à 60 mastocytes par mm^2 [Akimoto et al., 1998] de peau, ou 11 par mm^2 [Toyoda et al., 2000].

En plus de son rôle de déclencheur par libération d'histamine, le mastocyte joue aussi le rôle de cellule présentatrice d'Ag (peptide antigénique associé à une molécule du CMH de classe II). Lors de l'urticaire, le nombre de mastocytes est multiplié par 2,4.

Dans notre application, le modèle numérique du mastocyte est une organisation `Mastocyte` qui dérive de la classe `Cell`. Sa structure est composée de 22 espèces chimiques (voir fichier SBML en annexe A) et d'un corps. Elle est également composée d'instances de classes dérivées de `Phenomenon` : 2 phénomènes permettant la visualisation de l'état de la cellule et 16 phénomènes de type `ReactionPhenomenon` permettant la création des réactions chimiques

¹Nous pensons ici à l'influence de l'écoulement du plasma dans les tissus qui peut produire une pression sur la membrane et ainsi modifier la forme de la membrane.

²Nous utilisons ici beaucoup le conditionnel pour montrer ce qu'il est techniquement possible de réaliser avec notre outil. En général, c'est le manque d'information biologique pertinente qui a conduit à la non modélisation de tel ou tel aspect.

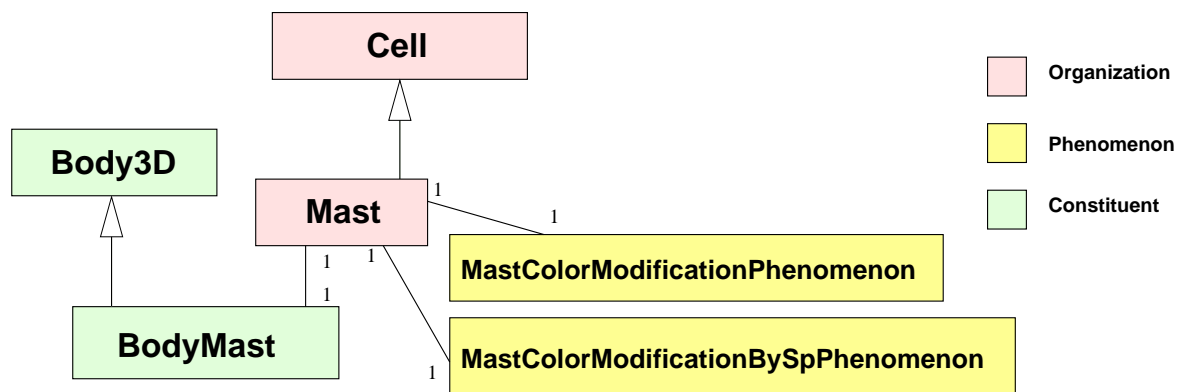


FIG. 6.6: Le corps de l'organisation Mastocyte.

qui régissent la dynamique de la cellule (ces derniers sont bien sûr générés automatiquement par l'interpréteur SBML).

Le corps des mastocytes est assez simple. La forme sphérique constitue en effet une approximation raisonnable de la cellule. Il est alors formé de deux sphères, la plus petite représente le noyau cellulaire (artefact esthétique), et l'autre la forme globale de la cellule. Ces formes 3D sont réparties aléatoirement dans l'espace du derme, c'est-à-dire dans toute la zone sous la membrane basale. Le diagramme de la figure 6.6 représente la classe *Mast*.

6.4.2 Relargage d'histamine et estimation de paramètres

Le relargage d'histamine par le mastocyte est la réaction la mieux connue du modèle. Deux produits présents dans le milieu (la substance P et l'allergène) vont permettre le relargage d'histamine. Il y a donc deux voies d'activation du mastocyte à modéliser.

Bien que la recherche bibliographique ait apporté de nombreuses informations, certaines, pourtant indispensables à l'établissement du modèle manquent. Il est donc apparu nécessaire d'estimer ces données. Pour cela, nous avons dû nous appuyer sur des résultats partiels, expérimentaux, obtenus dans la littérature, et les compléter pour les rendre utilisables.

Dans cette démarche, l'utilisation de la norme SBML se révèle très utile. L'idée est d'utiliser un logiciel tel CellDesigner (voir section 2.4.2) pour créer graphiquement un modèle chimique des voies qui nous intéressent. Ainsi, en fonction des résultats expérimentaux trouvés dans la littérature, le biologiste peut imaginer une cascade reproduisant la dynamique qui l'intéresse. La figure 6.7 montre les deux cascades produites. À l'aide du module de simulation de CellDesigner, il est possible de simuler le comportement d'une cascade avec des conditions initiales bien précises. Lorsque le comportement du modèle est satisfaisant, il est alors possible de récupérer le fichier SBML généré et de l'interpréter pour l'ajouter dans notre application multi-modèles. Les phénomènes décrits sont alors couplés avec le reste de la simulation.

Nous pouvons faire une remarque quant à la validité du modèle établi. Les modèles de réactions internes à la cellule ont été imaginés dans le respect des données partielles de la littérature et grâce à l'expertise du biologiste modélisateur. Ces données correspondent en général aux relevés de la concentration d'histamine en fonction du temps pour un stimulus donné. La première hypothèse est que le comportement global d'une population de n cellules correspond à n fois le comportement identique d'un individu. La deuxième hypothèse est que

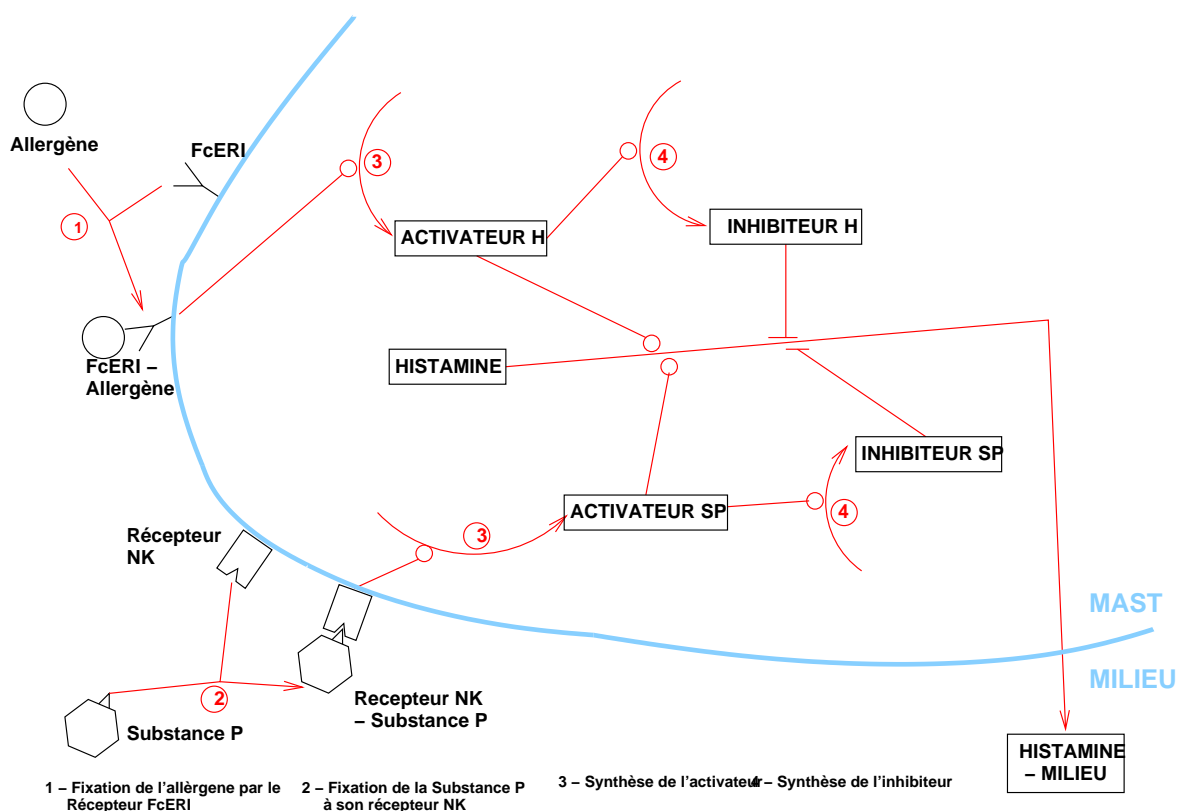


FIG. 6.7: Les deux voies principales d'activation du mastocyte.

la différence entre les conditions du modèle expérimental *in vitro* et celle du modèle *in vitro* n'invalide pas l'utilisation des résultats du premier pour la construction du second. Cette méthode n'est donc pas la panacée mais demeure la seule à notre disposition. En effet, n'ayant accès qu'aux données déjà existantes, nous devons nous fier à l'expérience du biologiste qui « imagine » des modèles réalistes de chaînes de réactions métaboliques se produisant dans la cellule. Ensuite, c'est l'expérience *in vitro* qui permet de faire l'expérience de ces modèles pour les affiner, les valider ou éventuellement les invalider.

6.4.3 Visualisation

Afin de visualiser les différents états d'activations du mastocyte, des interactions spécifiques ont été créées : `MastColorModification` et `MastColorModificationBySP`. La première intervient sur l'activation du mastocyte par l'allergène dans le milieu, et la seconde est liée à l'activation due à la fixation d'un neuropeptide appelé substance P sur les récepteurs du mastocyte. Lorsque le seuil d'activation est atteint, elle va entraîner la modification de la couleur du mastocyte, qui va passer du bleu au rouge pour l'interaction avec l'allergène, et vers le vert pour celle prenant en compte la substance P (voir figure 6.8).

Pour respecter la philosophie du modèle générique, l'interaction n'est pas directement instanciée par le modélisateur, mais une classe phénomène est créée pour chaque type d'interactions. Ainsi les interactions qui modifient la couleur seront créés automatiquement s'il y a lieu. Ces phénomènes ne sont que de simple artefacts de visualisation mais sont toutefois implémentés comme le reste du modèle, ce qui garantit l'homogénéité du code.

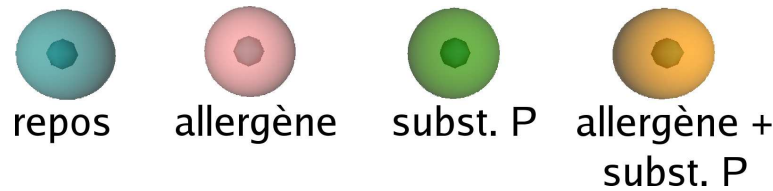


FIG. 6.8: Représentation 3D des différents états des mastocytes.

6.5 Capillaire sanguin

Nous avons choisi de modéliser les vaisseaux capillaires sanguins car ce sont eux qui, par augmentation de leur perméabilité, laissent s'écouler le plasma dans les tissus et provoquent l'apparition des symptômes observables de la réaction allergique.

Les capillaires de la peau sont composés d'une unique couche de cellules endothéliales³. Le diamètre des capillaires au niveau de la pupille est estimé à 10 - 12 μm [Braverman, 2000]. Une cellule endothéliale serait un pavé de 3 μm environ sur 6 - 7 μm de long.

Le capillaire sanguin, qui se développe à partir du tissu sous-cutané, traverse tout le derme, jusqu'à être au contact de l'épiderme. La densité en capillaire dans la peau est très variable, selon la zone étudiée [Pasyk et al., 1989]. L'ensemble des cellules endothéliales forme un vaisseau capillaire dans lequel circule le sang. Celui-ci permet de faire des échanges entre le milieu extracellulaire de cette zone et le reste de l'organisme (élimination des déchets, apports de nutriments, arrivée/départ de cellules).

Dans notre application, le modèle numérique du capillaire est une organisation **Capillary** composée d'un certain nombre d'organisations filles **EndothelialCell** dérivant de la classe **Cell** (voir figure 6.9). Un algorithme permet de générer aléatoirement une trajectoire pour le vaisseau capillaire qui est alors une succession de cellules endothéliales placées et orientées comme il faut (voir figure 6.10). Chaque cellule représente alors un tronçon du capillaire, ce qui correspond bien à la réalité biologique. Ainsi, le corps de ces cellules est représenté par un cylindre auquel est associée une sphère à chaque extrémité⁴.

Chaque organisation **EndothelialCell** est associée à une quantité de récepteurs H1 à l'histamine. La liaison des récepteurs avec l'histamine entraîne la production d'une espèce fictive, au sein de la cellule. Comme pour les mastocytes, cette espèce est associée à un phénomène modifiant la couleur du corps, ce qui permet la visualisation du niveau d'activation de la cellule (du rouge sombre au rouge clair). L'activation symbolise l'augmentation de la perméabilité du vaisseau.

L'augmentation de la perméabilité vasculaire a pour effet la libération de plasma dans le tissu avoisinant. Un nouveau type de phénomène a donc été créé pour permettre l'apparition d'une interaction **PlasmaReleasing**. Ainsi cette interaction unaire, « entre » un seul constituant (l'espèce fictive qui indique la perméabilité), a pour action la création de nouveaux constituants **Plasma**. Notons que l'interaction **PlasmaReleasing** est la première de ce type que nous rencontrons. Elle est particulière en ceci qu'elle ne se contente pas de modifier l'état

³Il n'y a pas de muscles lisses, ni tissu conjonctif à ce niveau. La vasodilatation n'est pas possible à ce niveau.

⁴Une astuce de modélisation consiste à considérer que le corps de la cellule n'interagit physiquement avec les autres corps que par l'intermédiaire des deux sphères des extrémités, ce qui évite de définir les collisions sphère/cylindre et triangle/cylindre. Le comportement du modèle reste adapté au besoin dans la mesure où la longueur de la cellule est faible.

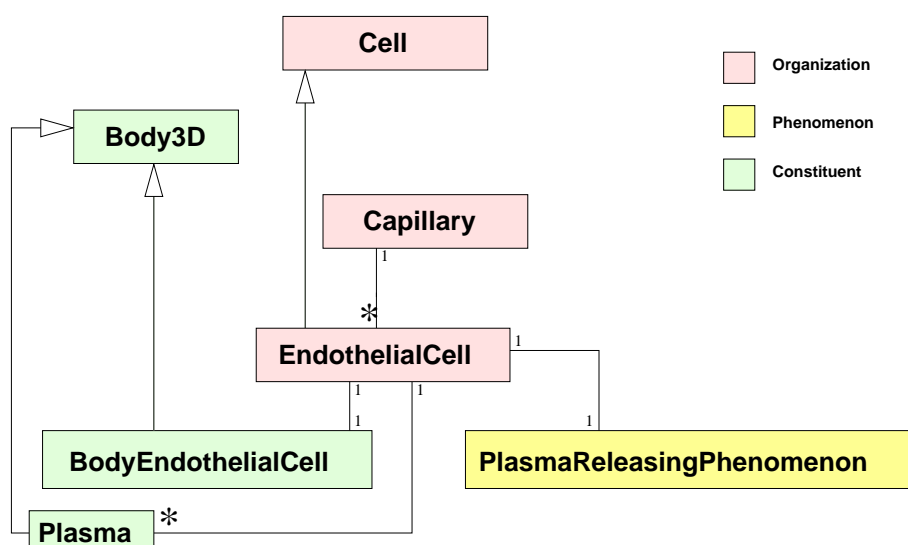


FIG. 6.9: Diagramme des classes modélisant le vaisseau capillaire.

de ses constituants mais qu'elle en produit de nouveaux.

Les constituants *Plasma* sont des corps qui modélisent un certain volume de plasma. Ils ont une masse très faible par rapport aux autres corps de la simulation. Si bien que lors d'une collision ils s'infiltreront autour des autres corps. Par contre, si le volume qui les contient est limité, on observe une augmentation de la pression sur la membrane basale. En rendant la membrane mobile (ce qui ne pose pas de problème technique particulier) on pourrait observer des déformations de l'épithélium.

L'œdème qui résulte de l'écoulement du plasma dans le tissu est le signe visible de la réaction de l'urticaire allergique. On voit alors comment il est possible de remonter depuis des considérations chimiques vers des effets observables à l'oeil nu. Malheureusement, les données concernant les volumes de plasma s'écoulant à travers un vaisseau capillaire perméable n'existent pas. Pour aller plus avant il faudrait donc les produire. Actuellement l'écoulement du plasma n'a donc qu'une fonction très qualitative.

6.6 Fibre nerveuse

Le derme est une zone riche en innervation. Il existe différents types de fibres : sensorielles (toucher, pression, sensation de chaleur...) et motrices. Celles-ci sont en général de faible diamètre et non myélinisées. Les fibres qui nous intéressent sont les fibres sensitives de type C, du système nerveux somatique. Ce sont elles qui vont intervenir lors de la réaction inflammatoire. Ce type de fibre est celui qui transmet le message nerveux à la vitesse la plus lente (de 0.5 à 2 m/s) [Ständer et al., 2003]. Le diamètre de ce type de fibre est estimé entre 0.4 et 1.2 μm . Ces fibres se divisent, pour former une réelle arborescence de terminaisons nerveuses. Elle sont le support de la transmission d'informations comme la douleur et le prurit. Ces deux sensations sont portées par des sous-populations différentes de fibres C. Le prurit peut être considéré comme un indicateur du déroulement d'une inflammation. Les fibres C sont activées lors de la réaction, et interviennent aussi lors de celle-ci. En effet, en plus de la transmission de messages, les fibres C vont relarguer dans le milieu extra-cellulaires des neuropeptides comme la CGRP ou la substance P, peptides jouant un rôle sur l'activation

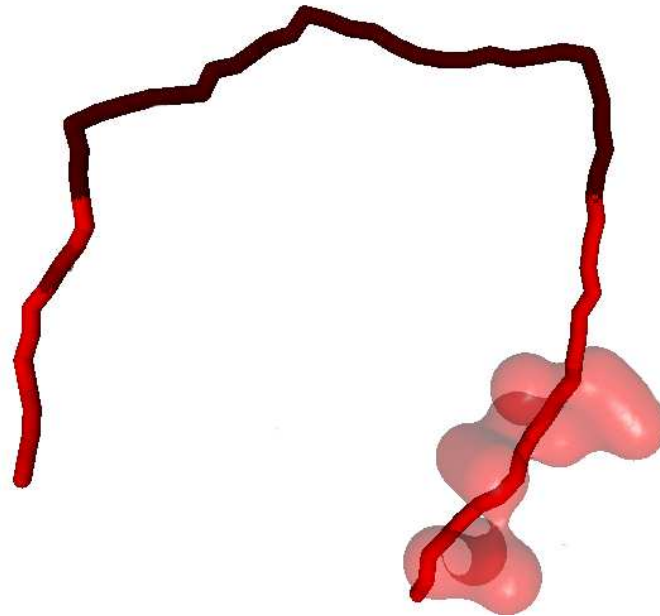


FIG. 6.10: Capillaire sanguin. Les zones plus claires sont perméables. À droite, on voit le plasma qui s'écoule.

mastocytaire [Pereira, 2005].

D'un point de vue modélisation, nous avons utilisé le même procédé pour modéliser la fibre nerveuse que pour modéliser le capillaire sanguin. Il existe tout de même une réelle différence conceptuelle entre le capillaire et la fibre nerveuse. En effet, le capillaire est un ensemble de cellules endothéliales, le découpage s'est donc révélé assez naturel à produire. La fibre nerveuse n'est elle, qu'une unique cellule, ce qui fait que la segmentation appliquée est plus artificielle. Ainsi, Les `NerveFiberPart` dérivent de `Cell` (voir figure 6.11) et composent l'organisation `NerveFiber`.

Il existe en plus une sous-partie particulière, appelée `NerveFiberEnding`, située à l'extrémité de la fibre, et qui possède un comportement particulier, en étant notamment à l'origine du relargage de substances dans le milieu. L'interaction spécifique à la fibre nerveuse

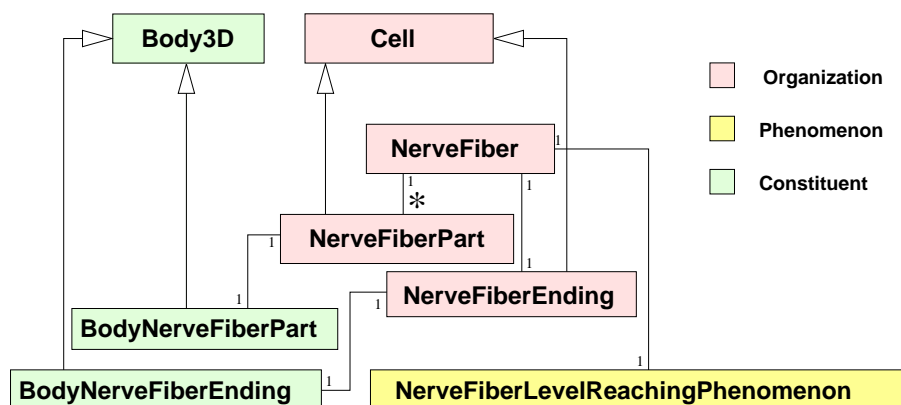


FIG. 6.11: Diagramme de classes de la classe `NerveFiber`.



FIG. 6.12: La fibre nerveuse : au repos (couleur grise) et activée (couleur jaune).

`NerveFiberLevelReaching` mesure à chaque activation la quantité de récepteurs activés et, une fois le seuil atteint, modifie la couleur des `BodyNerveFiberPart`, et active les relargages par le `NerveFiberEnding`. L'interaction `NerveFiberLevelReaching` reproduit cet effet « gâchette » d'activation de la fibre. Elle appartient non pas à `NerveFiberParts`, mais à `NerveFiber`, puisqu'elle fait appel à un mécanisme de seuil qui concerne l'ensemble de la fibre. La figure 6.12 présente la fibre nerveuse dans les deux états : au repos (couleur grise) et activés (couleur jaune).

6.7 Les interfaces

Bien qu'ayant choisi de ne pas approfondir le développement d'interfaces entre l'application et l'utilisateur dans cette étude, nous pouvons cependant en présenter quelques unes. Nos travaux s'inscrivant dans le cadre plus large des travaux du CERV, nous avons à notre disposition des moyens de visualisation d'univers en 3D fournis par ARéVi. Il est ainsi possible de se déplacer dans le modèle numérique, de l'observer au cours de l'exécution, voire de s'immerger dans le modèle (en utilisant des outils de stéréovision). L'utilisateur peut aussi agir sur le modèle, il peut modifier le modèle en déplaçant les formes 3D qui représentent les différents sous-modèles de cellules, de membranes ou d'organes. Enfin, à l'aide d'une seringue virtuelle, il peut injecter dans la peau une quantité d'une espèce moléculaire et en observer les conséquences.

6.7.1 Visualisation et interaction 3D

Le `Viewer3D` d'ARéVi permet la visualisation de tous les corps du modèle. Il est également possible d'y afficher d'autres informations tel le maillage de l'organisation chimique. Ensuite, il est possible de manipuler les corps dans l'espace. Ainsi l'utilisateur peut déplacer un cellule d'un endroit à l'autre du modèle qui s'adaptera de manière autonome. La figure 6.13 montre une capture d'écran du *viewer*. L'utilisation d'interfaces permettant l'immersion dans un univers en trois dimensions facilite la manipulation du modèle.

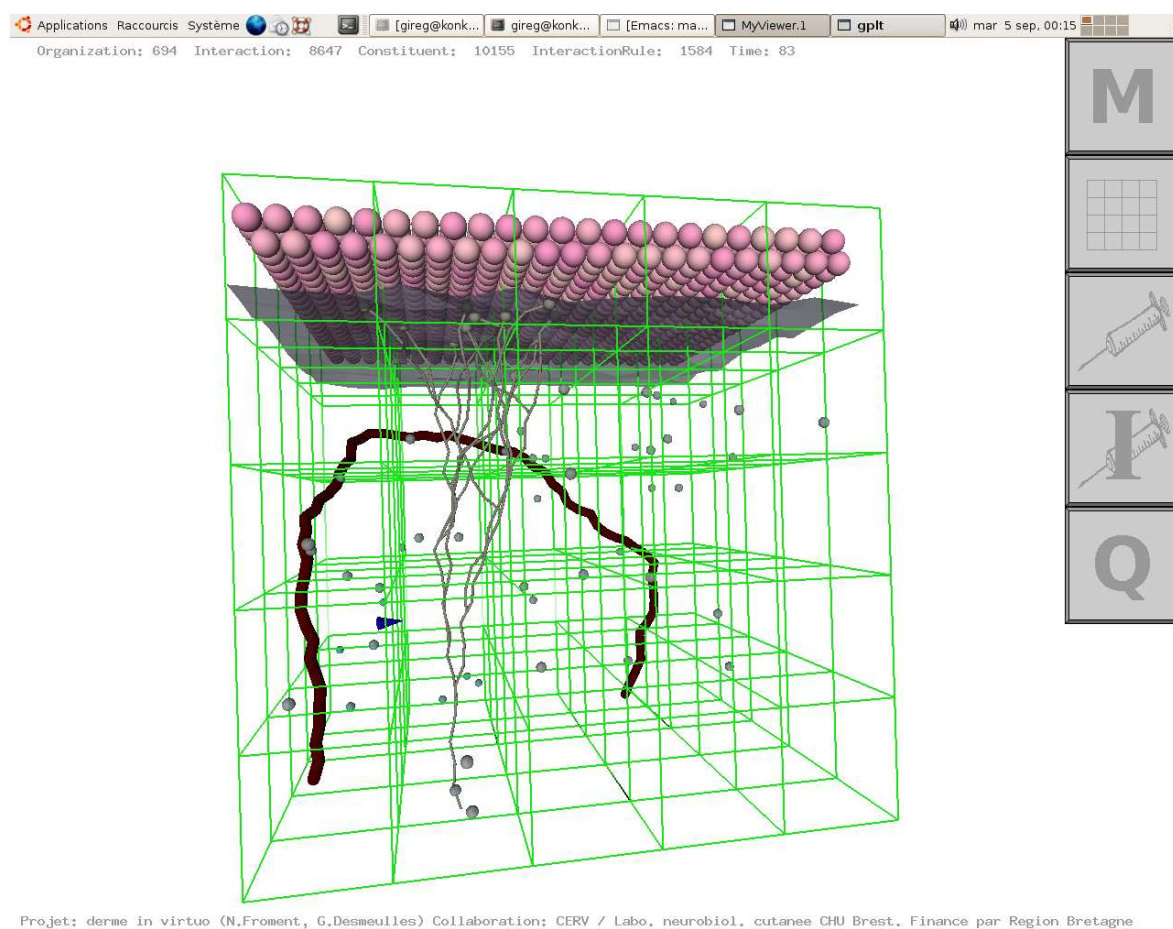


FIG. 6.13: Capture d'écran du viewer3D d'ARéVi pour l'application Derme *in virtuo*.

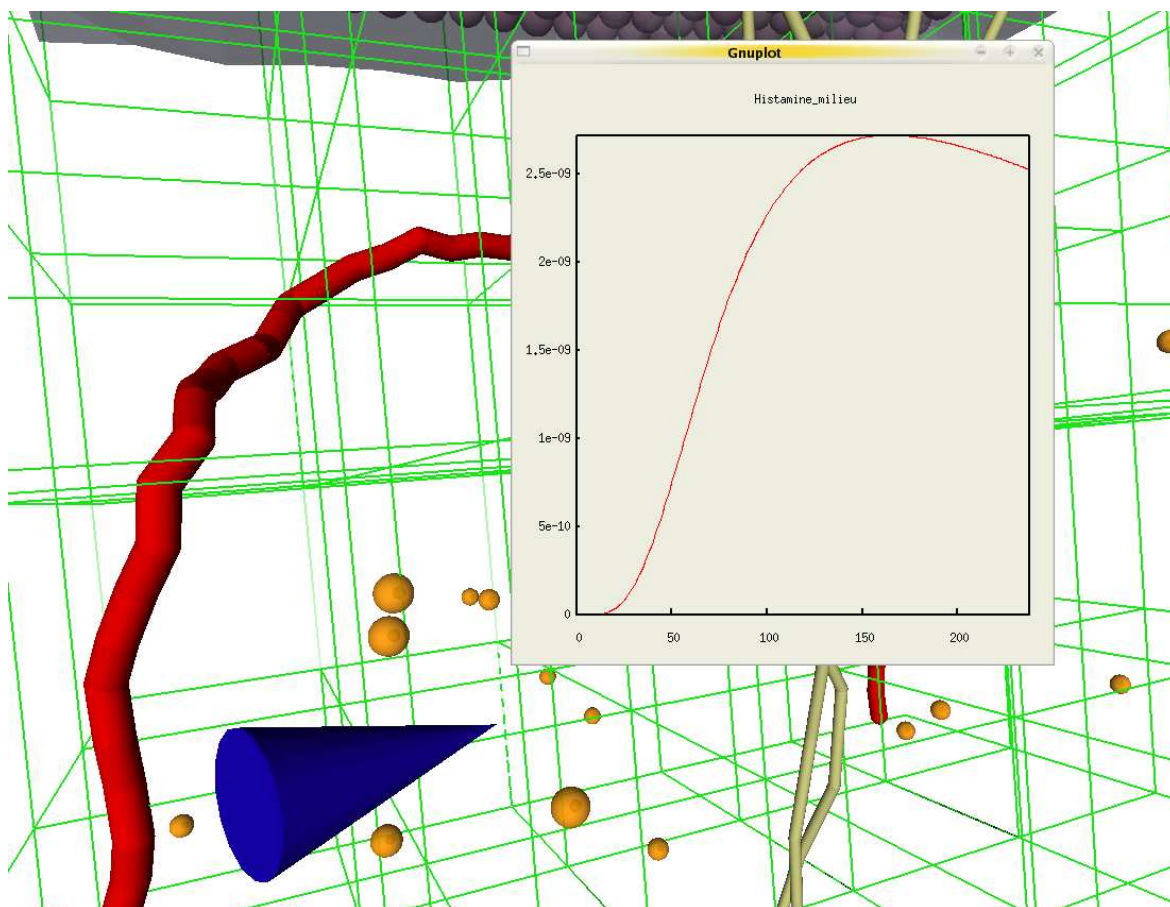


FIG. 6.14: La sonde représentée par le cône bleu, permet de visualiser les concentrations d'espèces chimiques à l'endroit pointé par celui-ci. Ici, nous voyons un tracé de la concentration d'histamine dans le milieu en fonction du temps (en secondes).

6.7.2 Sonde

L'utilisateur peut mesurer les concentrations d'espèces moléculaires à un endroit précis du milieu. Pour cela, il manipule une sorte de sonde qui inscrit sur un graphique la concentration de produit qui est présente à l'endroit pointé. La figure 6.14 montre une capture d'écran représentant la sonde.

6.7.3 Seringue

Le dernier moyen d'interfaces que nous présentons ici permet d'injecter une certaine quantité d'espèce chimique à un endroit du milieu. La seringue est un outil visuel qui permet à l'utilisateur de modifier l'état des constituants `Species` correspondant à un réacteur chimique de l'organisation chimique. La figure 6.15 montre l'expérience simple qu'il est possible de réaliser avec l'application.

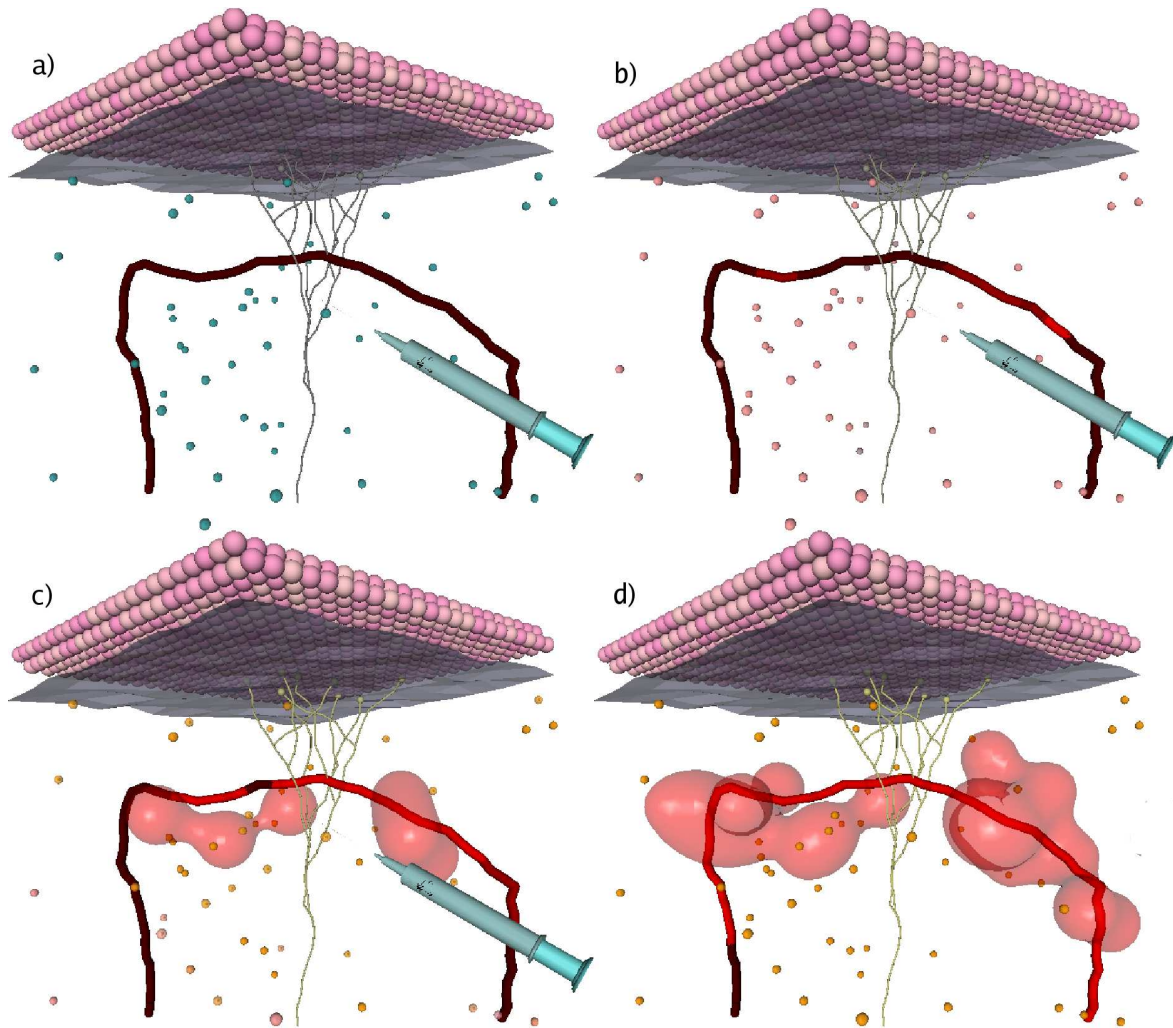


FIG. 6.15: a) La seringue permet d'injecter une certaine quantité d'allergène dans le milieu ; b) L'allergène active les mastocytes (qui passent du vert au jaune), ils relarguent alors de l'histamine dans le milieu, ce qui a pour effet d'activer les récepteurs du vaisseau capillaire ; c) L'histamine active la fibre nerveuse (qui change également de couleur), elle relargue alors un certain nombre de médiateurs qui amplifient l'activation des mastocytes qui relarguent encore plus d'histamine ; d) La perméabilité du vaisseau augmentant, du plasma s'écoule dans le tissu pour former un œdème. Il est possible de visualiser un film de cette expérience à l'adresse www.cerv.fr/fr/activites/EBV.php.

6.8 Discussions

Cette application est placée dans le contexte d'une collaboration scientifique avec des médecins et des biologistes. Elle permet de mettre concrètement en œuvre le cadre générique de modélisation que nous proposons. Nous nous confrontons ainsi aux attentes et aux contraintes posées par le domaine d'étude. La `libBio` nous a fourni une bonne expressivité pour créer les modèles numériques représentant l'état des connaissances. Nous avons modélisé le milieu du derme, la membrane basale, les mastocytes, le vaisseau capillaire et la fibre nerveuse. Nous sommes encore loin d'une validation du modèle biologique⁵. L'application permet cependant de formuler un certain nombre de remarques.

Tout d'abord, nous avons introduit dans le modèle une nouvelle composante aléatoire. Les positions initiales des cellules, les formes de la fibre nerveuse et celle du vaisseau capillaire sont des paramètres variables. Pour créer le modèle il faut nécessairement les définir de manière arbitraire (en interprétant tout de même les données bibliographiques). Pour tempérer cet aspect arbitraire et les biais qu'il introduisent, nous avons fait en sorte que ces éléments soient définis aléatoirement. Ainsi le nombre de ramifications de la fibre nerveuse, le chemin emprunté par ces ramifications, celui emprunté par le vaisseau, le placement et le nombre de mastocytes sont générés aléatoirement. A chaque nouvelle instanciation du modèle, les formes et positions initiales seront donc différentes. Cette composante aléatoire s'ajoute à celle introduite par l'ordonnancement chaotique. La première remarque est donc une remarque méthodologique : pour valider une hypothèse, il est nécessaire de multiplier les expériences *in virtuo*. C'est le procédé qui est déjà utilisé pour les expériences sur les systèmes réels.

La seconde remarque souligne que la construction du modèle se fait de façon incrémentale. À chaque fois qu'un nouvel élément est inséré dans le modèle il n'y a pas besoin de remettre en cause (informatiquement parlant) l'existant. Nous avons de plus constaté que la différence de natures des phénomènes modélisés n'était pas un problème. Ce confort dans l'implémentation est comme nous l'avons prédit, la conséquence de l'utilisation du principe d'autonomie. De plus, le fait d'inclure dans notre boîte à outil (`libBio`) tout ce qui peut être réutilisable nous amène à réduire en permanence le temps de développement de nouveaux aspects du modèle biologique. Nous affirmons d'ailleurs que le temps de développement du modèle numérique tend à devenir négligeable par rapport au temps nécessaire à l'établissement du modèle biologique.

En ce qui concerne les performances de la simulation, les tests réalisés sont encourageants. En effet, nous réussissons à maintenir la contrainte temps réel pour des modèles faisant vivre plusieurs milliers d'organisations et de phénomènes, et des dizaines de milliers d'interactions. De plus, l'ensemble des interactions contient plusieurs milliers d'instances de `Diffusion` et de `Reaction` qui possèdent des périodes d'activation de l'ordre du centième de seconde. La puissance d'un ordinateur domestique commence donc à être suffisante pour faire vivre des modèles de tailles significatives, ce qui laisse présager un bon avenir à ce type de modélisation.

Ensuite, nous pouvons insister sur le fait que ce travail applicatif ne représente qu'une étape dans l'établissement d'un modèle biologique, tout comme le modèle générique ne représente qu'une étape dans la construction de l'outil *in virtuo*. Nous devons concéder que le

⁵Peut-être est-ce parce que l'état des connaissances ne permet pas aujourd'hui d'établir un modèle explicatif beaucoup plus détaillé de l'urticaire allergique.

mode d'expérimentation reste encore limité car le travail sur les interfaces n'a pas encore été réalisé (voir section 5.4). De même, le modèle biologique n'est pas abouti. Il évoluera au fil des collaborations : chaque rencontre avec un spécialiste, chaque nouvelle publication trouvée permet d'étoffer un peu plus le modèle.

La principale limite à l'heure actuelle réside dans le manque de données biologiques. Certaines des données qui nous font défaut existent, mais sont la propriété de laboratoires privés et ne seront accessibles que par l'intermédiaire de collaborations avec des partenaires privés. Notons également que d'autres données manquent car elles peuvent ne pas sembler pertinentes dans l'approche biologique classique alors qu'elles présenteraient un intérêt pour la modélisation. Ainsi, nombre de paramètres, de constantes cinétiques... ne sont pas publiés alors que leur établissement ne pose pas de problèmes techniques particuliers. Pour avancer, il faudrait alors des biologistes à la paillasse qui produisent les données pour compléter les modèles.

Nous voyons apparaître deux phases dans l'établissement d'un modèle numérique :

1. Recherches bibliographiques et établissement d'un modèle biologique *in virtuo* correspondant à un état des connaissances cohérent du sujet d'étude.
2. Expérimentation (et donc éaction) du modèle numérique qui entraîne la production de nouvelles hypothèses qui peuvent être testées sur le modèle numérique ou sur le modèle physique pour aboutir à la production de nouvelles connaissances complétant le modèle.

Ces considérations sont présentes dans tout travail de modélisation. Nous souhaitons simplement fournir un mode d'expérimentation des modèles plus riche et donner un champ d'expression plus large au modélisateur. Ajoutons qu'à tout moment les vertus pédagogiques du modèle *in virtuo* sont considérablement utiles pour la collaboration entre chercheurs.

Finalement, indiquons que ces travaux suscitent l'intérêt de la communauté des dermatologues puisqu'ils ont été sujet à publications [Desmeulles et al., 2003a,b, 2005d,a]. Nous les présentons ici comme une illustration du modèle générique proposé dans la thèse. Le chapitre suivant a pour but de compléter cette illustration en exposant deux applications en cours de développement montrant l'expressivité de notre modèle générique.

Chapitre 7

Figures de style

Ce court chapitre présente deux applications supplémentaires dont les principes permettent de montrer d'autres possibilités du modèle générique. Par simple manque de temps, la première de ces applications est malheureusement inachevée et la seconde assez modestement approfondie. Il aurait toutefois été dommage de se priver de l'éclairage supplémentaire que leur conception nous apportent.

En premier lieu, nous décrivons un projet de maillage de l'espace pour la diffusion moléculaire. La méthode permet aux modèles numériques peuplant l'univers virtuel de structurer eux-mêmes leur environnement, ce qui présente un pas supplémentaire vers l'autonomisation des modèles et vers l'allègement des calculs.

Deuxièmement, nous exposons un projet de modélisation de systèmes autopoïétiques imaginaires inspirés du fonctionnement multi-cellulaire. En posant de nouvelles questions, cette démarche permet d'élargir la discussion à la modélisation de phénomènes telle la reproduction.

7.1 Un autre modèle de diffusion

Dans l'application « *Derme in virtuo* », la simulation des phénomènes chimiques est la principale consommatrice en puissance de calcul. Afin d'augmenter la taille des modèles, nous cherchons des solutions permettant de réduire son coût. Pour se faire, nous réutilisons un principe qui a déjà été mis en œuvre lors de travaux sur la modélisation énaactive de la mer [Parenthoën and Tisseau, 2005]. L'idée est de permettre aux modèles numériques de structurer eux-mêmes l'univers dans lequel ils évoluent, pour que ne soient effectués que des calculs utiles.

Dans cette section, nous exposons tout d'abord, le problème de la structuration prédéfinie du milieu. Puis, nous proposons une solution utilisant des plans médians. Nous intégrons cette solution dans le formalisme de notre modèle générique, puis nous présentons la solution technique retenue. Nous formalisons enfin les calculs pour la diffusion.

7.1.1 Le problème de la structuration *a priori*

Telle que nous l'avons définie dans la section 5.1.4, l'organisation chimique découpe l'environnement en milieux homogènes cubiques. Toutes les mailles possèdent alors une taille identique. Cette taille doit correspondre à la plus petite nécessaire pour opérer une ap-

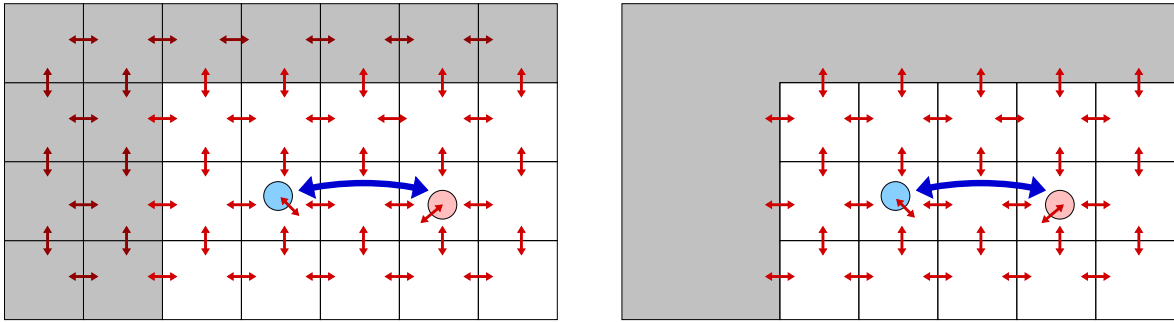


FIG. 7.1: Deux cellules communiquent *via* l'émission et la réception de médiateurs chimiques dans un milieu discrétisé. L'interaction entre les deux cellules (représentée par la flèche bleue) se matérialise par l'action de l'ensemble des interactions de diffusion entre les compartiments (flèches rouges). Le modèle de droite est une bonne approximation de celui de gauche si on ne se focalise que sur l'interaction entre les deux cellules.

proximation acceptable des phénomènes de transport durant toute l'expérience *in virtuo*. Ce maillage homogène a alors pour conséquence le maillage inutilement précis de certaines régions du modèle. La figure 7.1 illustre cette affirmation.

Dans l'organisation chimique, une maille correspond à une organisation **Reactor** qui est associée à un constituant **Compartment**, à plusieurs constituants **Species** et composée de toutes les interactions **Reaction** qui accomplissent les transformations chimiques du milieu. Une maille est donc un modèle assez lourd en soi, d'autant qu'entre chaque maille il faut ajouter les interactions réalisant la diffusion. Or, à chaque fois que nous voulons augmenter la précision du maillage nous augmentons le nombre de mailles. Comme nous travaillons dans un univers en 3 dimensions, la taille du modèle grandit en fonction du cube de la précision. Ainsi, dans l'application « *Derme in virtuo* », pour un pas de $0,2\text{mm}$ nous avons 125 réacteurs chimiques, pour un pas de $0,1\text{mm}$ nous en avons 1000. Au delà, la taille du modèle devient trop importante pour respecter la contrainte temps réel.

Dans certains cas, il est possible de vouloir modéliser au niveau cellulaire des mécanismes de perception de gradient de concentration qui porte sur des distances inférieures à $0,1\text{mm}$. Avec un maillage homogène, il devient rapidement inenvisageable de mettre en place un maillage suffisamment fin. La solution paraît être alors de définir un maillage non homogène, plus fin dans les régions critiques et plus lâche dans les régions où il ne se passe rien. Malheureusement, le comportement autonome des entités qui peuplent le modèle, et le caractère expérimental de notre approche rendent inadaptée l'idée de définir un maillage *a priori* puisque nous nous interdisons de pré-supposer quelles seront les régions critiques. La solution consiste alors à déléguer la structuration du maillage du modélisateur vers les modèles. Nous cherchons alors un procédé aboutissant à un maillage non homogène, adaptatif et auto-adapté.

7.1.2 Maillage par plans médians

Lorsque nous modélisons le phénomène de la diffusion, nous modélisons un phénomène de transport. Les espèces moléculaires transportées ont en général un rôle de communication : voies de communication extra cellulaires (par exemple par le relargage d'histamine), ou un rôle d'alimentation en énergie. Il y a donc toujours un ou plusieurs producteurs ou émetteurs, et un ou plusieurs consommateurs ou récepteurs. Lorsque nous modélisons la diffusion, nous modélisons indirectement l'interaction entre émetteurs et récepteurs. Nous devons donc nous assurer qu'entre ces deux types d'acteurs, le maillage permet la mise en œuvre adaptée du phénomène de transport.

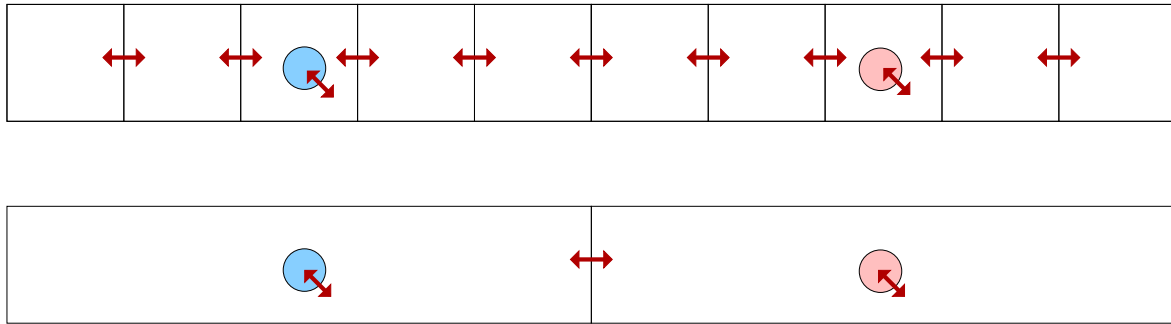


FIG. 7.2: Deux cellules communiquent *via* l'émission et la réception de médiateurs chimiques dans un milieu discrétisé. Moyennant une approximation raisonnable, les modèles du dessus et du dessous sont équivalents si on ne se focalise que sur l'interaction entre les deux cellules. La seule diffusion à travers le plan médian suffit à réaliser le phénomène qui nous intéresse.

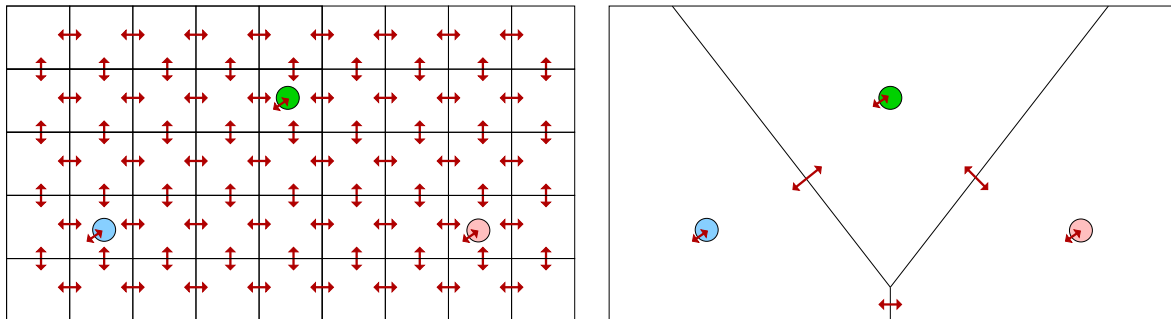


FIG. 7.3: Trois cellules communiquent *via* l'émission/réception de médiateurs chimiques dans un milieu discrétisé. Le modèle de droite est certes moins précis, mais il constitue une approximation adaptée.

Concrètement, nous raisonnons ici sur la diffusion dans un milieu permettant la simulation d'un système multi-cellulaire. Les émetteurs/récepteurs sont alors incarnés par des organisations `Cellule`. Si on s'intéresse à deux cellules, la seule diffusion à travers un plan médian suffit à réaliser le rôle de transport de l'information (voir figure 7.2). L'idée est alors de généraliser ce principe en ne prenant en compte que les plans médians entre les cellules. Dans l'exemple en deux dimensions de la figure 7.3, la taille du modèle chimique est divisée par 25, moyennant une approximation qui reste acceptable.

Cherchons maintenant à intégrer ce nouveau point de vue dans notre modèle.

7.1.3 Intégration dans le cadre générique

L'idée est ici de redéfinir l'organisation chimique (`ChemicalOrganization`) définie à la section 5.1.4 en l'adaptant à la nouvelle manifestation du phénomène de la diffusion à travers les plans médians. Rappelons que son principe initial était de créer un réacteur chimique par cube d'espace, l'objet `DiffusionPhenomenon` se chargeant d'instancier les interactions `Diffusion` entre les constituants `Species` et `Compartment` connexes.

Ici, il ne faut créer qu'un seul réacteur chimique initial dont le compartiment recouvre l'ensemble de la région que l'on souhaite modéliser. Le réacteur initial est toujours de type `ModelSBML`. Ensuite, chaque fois qu'un autre système autonome souhaite interagir avec le milieu, il crée un compartiment « milieu ». L'organisation chimique insère ce nouveau compartiment dans sa topologie, se faisant, le phénomène de diffusion, suivant la méthode choisie,

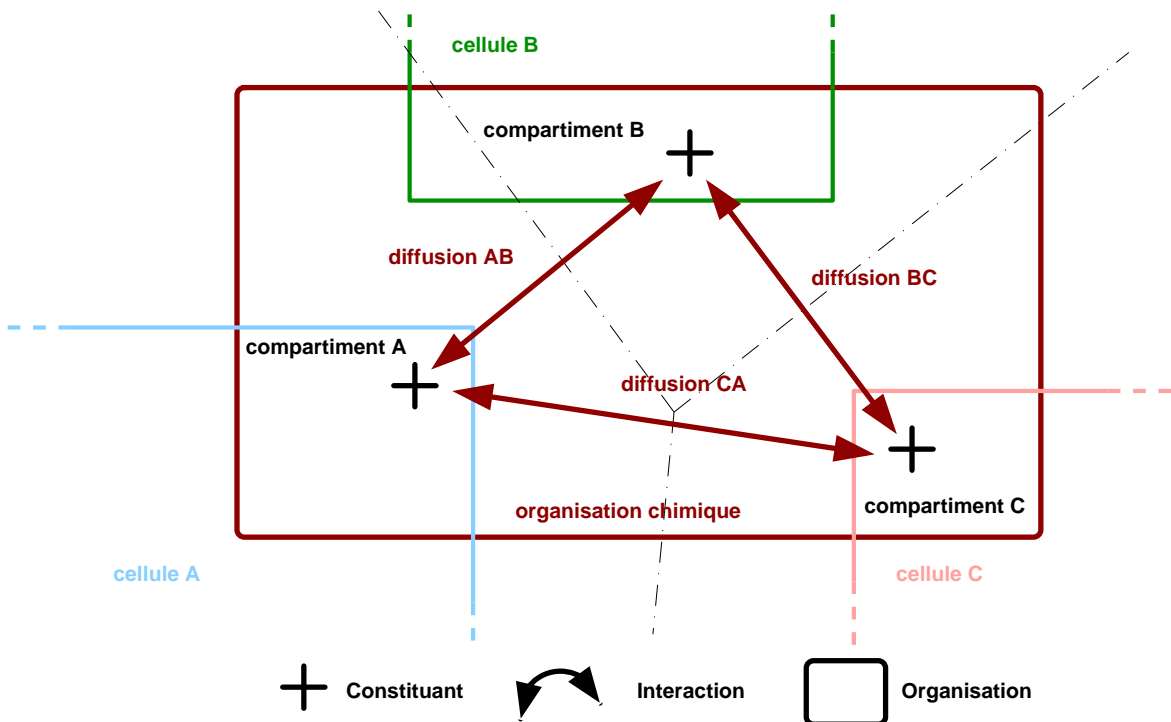


FIG. 7.4: L'organisation chimique est couplée avec les cellules A, B, C. Les espèces ne sont pas représentées.

réalise un maillage affectant à chaque compartiment un volume particulier. Il devient possible d'instancier des interactions **Diffusion** entre tous les compartiments « milieu » connexes.

La figure 7.4 représente alors l'organisation chimique de la même façon que la figure 5.8 représentait l'organisation mécanique. Ensuite, la figure 7.5 montre un exemple plus compliqué d'organisation chimique. Le principe est toujours le même : chaque système évolue de manière autonome en couplage structurel avec les autres. La description du phénomène de diffusion est alors cantonné à l'organisation chimique. Plus particulièrement, la méthode de maillage est implémentée dans le phénomène **Diffusion**. Nous la décrivons ici, plus en détail.

7.1.4 Solution technique : diagramme de Voronoï

D'un point de vue technique, chercher à partitionner l'espace en construisant les plans médians entre des points de l'espace correspond à la construction d'un diagramme de Voronoï en 3D [Okabe et al., 2000]. Il existe de nombreux travaux sur les diagrammes de Voronoï, cependant la plupart d'entre eux traite des problèmes en 2D. Le nombre d'outils offrant des possibilités de manipulation de diagrammes de Voronoï en 3D est lui plus restreint car la génération de tels diagrammes revêt une certaine complexité algorithmique. Il y a au moins deux façons de décrire le principe du maillage en cellules de Voronoï.

En premier, nous considérons le diagramme de Voronoï comme une décomposition particulière d'un espace métrique déterminée par les distances à un ensemble discret d'objets de l'espace, ici un ensemble discret S de points. On appelle cellule de Voronoï associée à un élément p de S l'ensemble des points qui sont plus proches de p que de tout autre point de

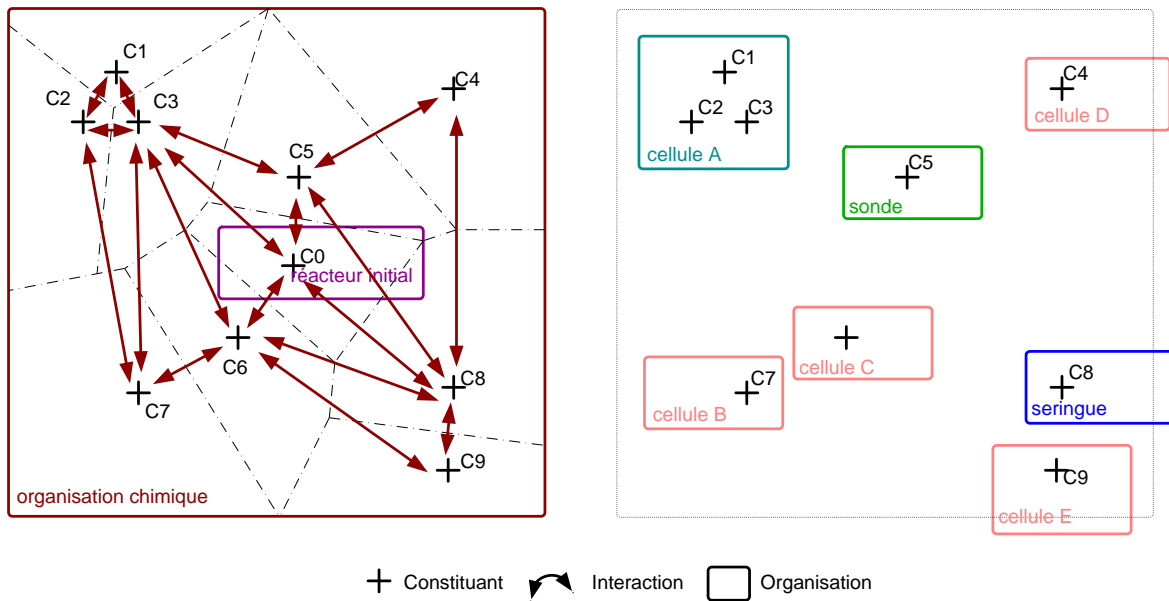


FIG. 7.5: À gauche, l'organisation chimique. À droite, les systèmes avec lesquels elle est couplée.

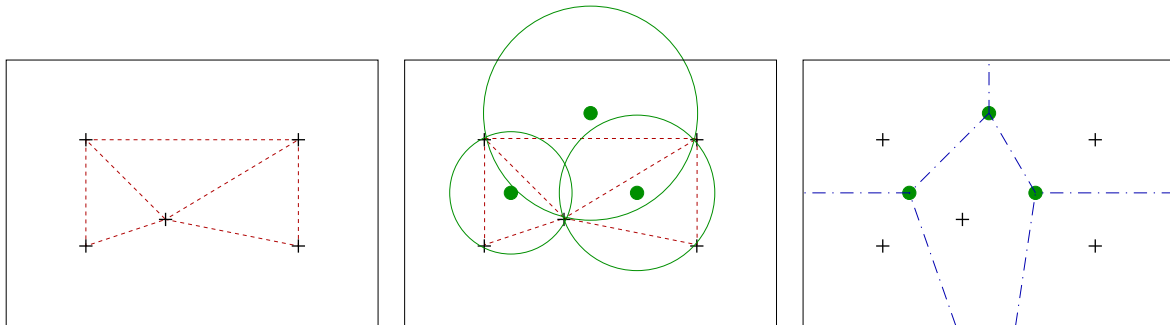


FIG. 7.6: On commence par établir la triangulation de Delaunay d'un nuage de points (à gauche). Ensuite, on définit l'isobarycentre de chaque triangle (au centre). Enfin, on crée les cellules de Voronoï à partir des sommets trouvés (à droite).

S . Une cellule de Voronoï est alors un polytope¹ convexe, obtenu en faisant l'intersection des demi espaces formés par les plans médians entre chaque point de S .

La seconde façon d'appréhender notre problème est de considérer qu'un diagramme de Voronoï est dual à une triangulation de Delaunay. En effet, chaque sommet d'une cellule de Voronoï, associé à un point p , correspond à l'isobarycentre de chaque triangle de Delaunay auxquels appartient p (voir figure 7.6). Or, l'usage de la triangulation de Delaunay est très répandu en informatique graphique. Parmi les outils la mettant en œuvre, l'application du projet CGAL² est probablement un des plus rapide.

Ainsi, le phénomène **Diffusion** opère une triangulation de Delaunay sur les positions de l'ensemble des compartiments « milieu », en utilisant la bibliothèque CGAL. Ensuite, il est possible d'obtenir immédiatement l'ensemble des points constituant les sommets de chaque cellule de Voronoï. Cet ensemble de points doit ensuite être transformé en polyèdre. Les opérateurs booléens de CGAL permettent de calculer les intersections entre anciennes

¹Polygone en 2D, polyèdre en 3D...etc

²www.cgal.org

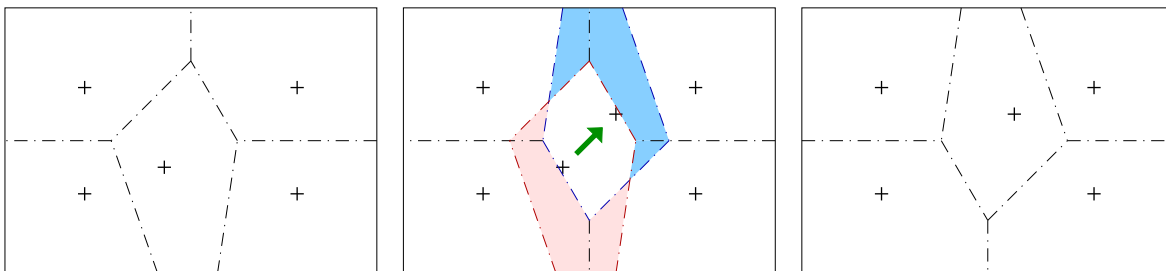


FIG. 7.7: Lors du déplacement d'un point dans le diagramme, la forme des cellules de Voronoï change. En calculant les intersections entre les anciennes et les nouvelles cellules, on peut établir les quantités de molécules dans chaque compartiment. Ici, lors du déplacement du point central, les quantités de molécule correspondant aux régions roses sont ajoutées aux compartiments connexes et les quantités correspondant aux régions bleues sont ajoutées au compartiment central; pour cela nous devons déterminer le volume de chaque région.

et nouvelles cellules de Voronoï pour répartir les quantités d'espèces moléculaires dans les compartiments, lors de mouvements ou de la création d'une cellule de Voronoï (voir figure 7.7).

La librairie CGAL est très efficace quant à la réalisation de la triangulation de Delaunay et du diagramme dual de Voronoï. L'établissement de ce diagramme représente en théorie la partie la plus difficile algorithmiquement. Cependant, il apparaît que les outils de manipulation de polyèdres de CGAL sont trop génériques pour notre cas simple de création de polyèdres convexes et du calcul du volume de leurs intersections. Nous n'avons pas encore réussi à mettre en place un algorithme suffisamment efficace pour calculer les intersections et les volumes des intersections de polyèdres. Nous pensons toutefois qu'il ne s'agit là que d'une simple question de temps de développement. Définissons alors le comportement de l'interaction Diffusion qui utilisera le maillage.

7.1.5 Formalisation du calcul de l'interaction Diffusion

La première loi de Fick nous indique que le flux \vec{j} en $mole.s^{-1}$ à travers une surface S en m^2 est donné par l'équation :

$$\vec{j} = -D.S.\vec{grad}(C)$$

où $C(x, y, z)$ est la concentration de l'espèce en un point donné, en $mole.L^{-1}$ et D le coefficient de diffusion en $m^2.s^{-1}$.

Nous intégrons le flux entre les points qui nous intéressent, c'est-à-dire le centre des cellules de Voronoï, puis faisons l'approximation que la surface de diffusion et le gradient de concentration sont constants entre ces deux points. Nous obtenons ainsi la quantité ΔQ (en $mole$) de matière à passer d'un compartiment chimique à l'autre :

$$\Delta Q = -D.S.\frac{\Delta C}{\Delta x}.\Delta t$$

où S est la surface de contact entre les deux cellules de Voronoï, Δt le pas de temps de l'interaction, ΔC la différence de concentration entre les deux compartiments et Δx la distance entre les deux centres des cellules.

Remarquons que si nous nous intéressons au cas particulier de mailles cubiques, alors

$$S = \Delta x^2$$

$$\Delta Q = -D \cdot \Delta x \cdot \Delta C \cdot \Delta t$$

ce qui correspond à l'équation utilisée à la section chapitre 5.1.2.

7.1.6 Perspectives

Au delà de la simple optimisation d'un phénomène, ce qu'il faut retenir c'est que ce sont les modèles qui structurent eux-mêmes l'environnement en créant du milieu là où ils souhaitent subir les perturbations de leur environnement. Chaque modèle décide ainsi du maillage adapté. C'est le fait de construire nos modèles comme un assemblage de systèmes autonomes en couplage structurel qui permet cette approche.

Notons tout de même qu'il est difficile de valider la méthode sans l'avoir encore testée, d'autant qu'il est également difficile de quantifier l'erreur sur un maillage irrégulier car il faudrait d'abord quantifier l'irrégularité du maillage. Avant de se lancer dans une validation mathématique équivalente à celles déjà réalisées pour les réactions chimiques et la diffusion sur un maillage cubique, il faudra comparer la méthode avec d'autres qui sont déjà validées, sur des cas concrets d'utilisation. Nous avons cependant bon espoir d'obtenir des résultats avec cette méthode qui permet d'adapter automatiquement le maillage en cours de simulation. L'ensemble interaction `Reaction` et interaction `Diffusion` fournira alors un moyen adapté et le moins coûteux possible pour intégrer des mécanismes de réaction/diffusion dans nos univers multi-modèles et temps réel.

Pour l'instant nous adaptons le maillage lors du déplacement d'un compartiment, par suppression/ajout d'un point dans la triangulation. Nous pouvons citer de récents travaux [Ledoux and Gold, 2006] sur la modélisation de données océanographiques qui visent à établir un algorithme permettant l'utilisation de points flottants dans les diagrammes en 3D. Ainsi, les calculs pourraient s'en trouver considérablement allégés.

Enfin, retenons qu'il est intéressant de développer les outils de maillage par diagramme de Voronoï pour l'application « derme *in virtuo* » puisque la géométrie de la membrane des kératinocytes dans l'épiderme est souvent modélisée par les plan médians aux noyaux.

7.2 Un système autopoïétique minimal

7.2.1 Systèmes vivants et systèmes du vivant

L'objet de la modélisation de l'outil que nous proposons dans cette thèse est l'étude du vivant. Pour le concevoir, nous nous sommes inspirés des travaux de Varela qui proposent une démarche visant à rendre intelligibles les systèmes vivants. La méthode repose notamment sur les notions de clôture opérationnelle, de systèmes autonomes en couplage structurel et de systèmes autopoïétiques.

Plus concrètement, nous avons appliqué notre modèle générique à la modélisation d'un phénomène partiel constitutif de l'organisme humain : la réaction allergique (voir chapitre 6). Nous n'avons pas eu besoin de la propriété autopoïétique que Varela définit comme une propriété qualifiant par définition un système vivant. Autrement dit, nous avons appliqué notre modèle à la modélisation d'un système du vivant, mais pas à la modélisation d'un système vivant. Cette dernière section a donc pour but de montrer comment il est possible de définir

et de faire vivre un système autonome autopoïétique à partir de notre modèle générique. Nous pourrions alors soulever quelques questions quant à la dynamique des organisations.

7.2.2 L'autopoïèse

Le terme autopoïèse vient du grec *auto* (soi-même), et *poiësis* (production, création). Il définit la propriété d'un système à se produire lui-même. L'approche autopoïétique de Maturana et Varela est née à Santiago du Chili. Faisons une nouvelle fois référence au livre « Autonomie et connaissance » [Varela, 1989a] : *un système autopoïétique est organisé comme un réseau de processus de production de composants qui*

- *régénèrent continuellement par leurs transformations et leurs interactions le réseau qui les a produits*
- *constituent le système en tant qu'unité concrète dans l'espace où il existe, en spécifiant le domaine topologique où il se réalise comme réseau.*

Chaque être vivant est un exemple de système autopoïétique. La cellule est peut-être celui que l'on cite le plus souvent. La cellule humaine par exemple est composée de protéines d'ADN, d'ARN et d'un certain nombre de molécules organisées dans des structures tels le noyau, les membranes, les organites... Ces structures sont produites par la cellule elle-même à partir du flux de molécules et d'énergie qui la traverse. Elles constituent de plus la machinerie qui permet de les maintenir et les produire. La cellule contient les processus de sa propre production et de son maintien. Elle est un système autopoïétique.

Comme le modèle générique le définit, la classe `Organisation` permet de décrire des systèmes autonomes (voir section 4.3). La distance qui sépare un système autonome d'un système autopoïétique est la présence à la fois de la propriété homéostatique et de celle d'autoproduction. Donc, pour définir un système autopoïétique nous devons définir une organisation ayant comme *but* le maintien de l'homéostasie³ du système qu'elle définit et la doter de processus de production. C'est l'exercice que nous nous proposons de réaliser.

7.2.3 Système proposé

7.2.3.1 Structure dissipative

En premier lieu, nous choisissons de mettre en œuvre une métaphore d'organisme multicellulaire, ce qui nous permet de réutiliser les outils de la `libBio` que nous avons déjà utilisés pour la modélisation de la peau. Ainsi, nous commençons par utiliser l'organisation `ChemicalOrganization` (voir section 5.1.4) pour définir le milieu dans lequel évoluera notre système.

De plus, l'autopoïèse fait référence à la dynamique des structures en équilibre instable, c'est-à-dire aux structures dissipatives. Les structures dissipatives sont des ensembles d'états organisés qui restent stables pour de longues périodes en dépit de la matière et de l'énergie qui passent à travers [Prigogine and Stengers, 1997]. Le tourbillon qui se forme lorsqu'on vide une baignoire est un bon exemple de structure dissipative. Le tourbillon est un *pattern* dynamique qui perdure grâce au flux de matière qui le traverse et le compose. Ce flux est le résultat de la dissipation de l'énergie potentielle des molécules d'eau qui s'écoulent dans

³L'homéostasie est la capacité à conserver l'équilibre de fonctionnement en dépit des contraintes extérieures. C'est un terme initialement utilisé en biologie qui peut cependant s'appliquer à une variété de systèmes autorégulés.

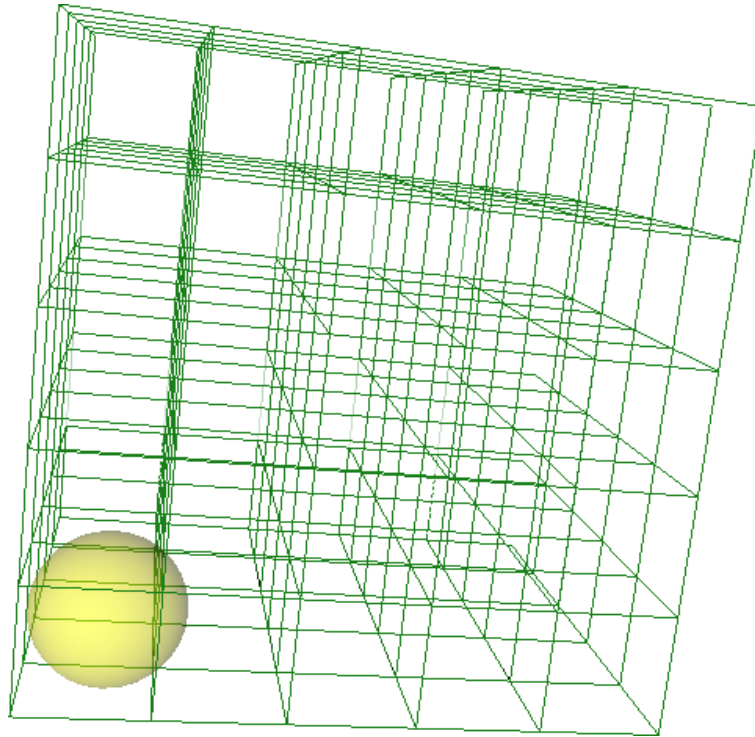


FIG. 7.8: Capture d'écran de l'environnement du système autopoïétique. La sphère jaune représente une source d'énergie.

le siphon. Ainsi pour mettre en œuvre un système autopoïétique, il nous faut ajouter une source d'énergie dans notre baignoire représentée par le milieu chimique. Nous créons alors un objet de type `SourceEnergie` héritant de la classe `Cell`, qui déverse une certaine quantité d'énergie dans le milieu. Nous nommons cette énergie « glucose ». La figure 7.8 montre une capture d'écran du milieu chimique discrétisé et de la source d'énergie. La figure 7.9 décrit le principe de l'organisation `SourceEnergie`.

7.2.3.2 La brique élémentaire `MyCell`

Nous avons choisi de mettre en œuvre un système multicellulaire. La brique de base de notre système est donc la cellule. La cellule est certes elle-même un système autopoïétique dans la nature, mais ici, nous ne la considérons qu'en tant que système autonome pouvant se reproduire et servant à construire un système autopoïétique d'ordre supérieur.

La classe `MyCell` dérive alors de l'organisation `Cell` (voir figure 7.10). Nous lui associons un fichier SBML décrivant les phénomènes chimiques qui s'y réalisent. La cellule absorbe le « glucose » en le transformant en « énergie ». L'« énergie » est alors consommée. La consommation a pour effet de produire un déchet nommé « régulateur ». Cette dernière espèce se dégrade naturellement dans le milieu. Ensuite nous avons ajouté deux phénomènes agissant sur le comportement du système : la mitose et l'apoptose (voir figure 7.11).

La mitose (division cellulaire) est le processus de production qui sera nécessaire à l'autopoïèse du système de second ordre que nous présentons. L'avantage de la modélisation est ici de permettre la factorisation de tous les mécanismes extrêmement complexes aboutissant à la reproduction cellulaire en un simple phénomène `MitosePhenomenon`. L'interaction `Mitose` augmente la concentration de l'espèce « mitoseActivateur » au cours du temps. Ensuite, le

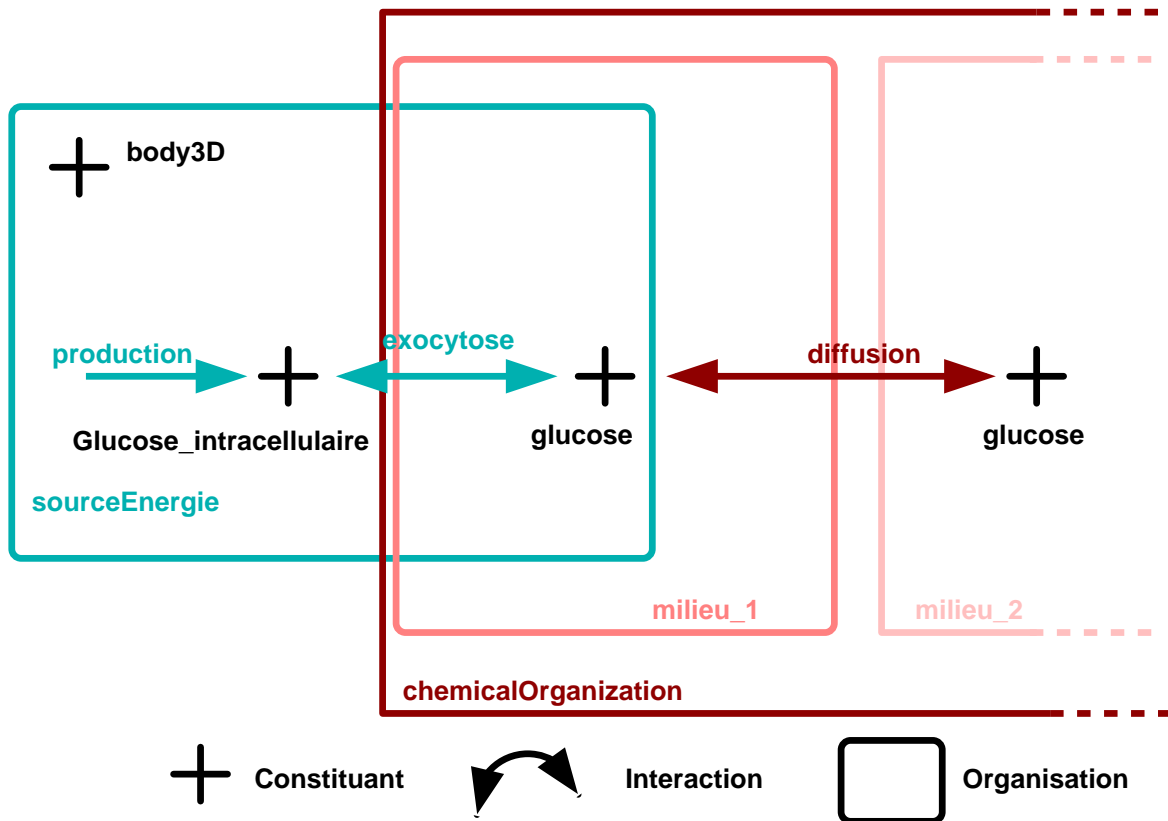
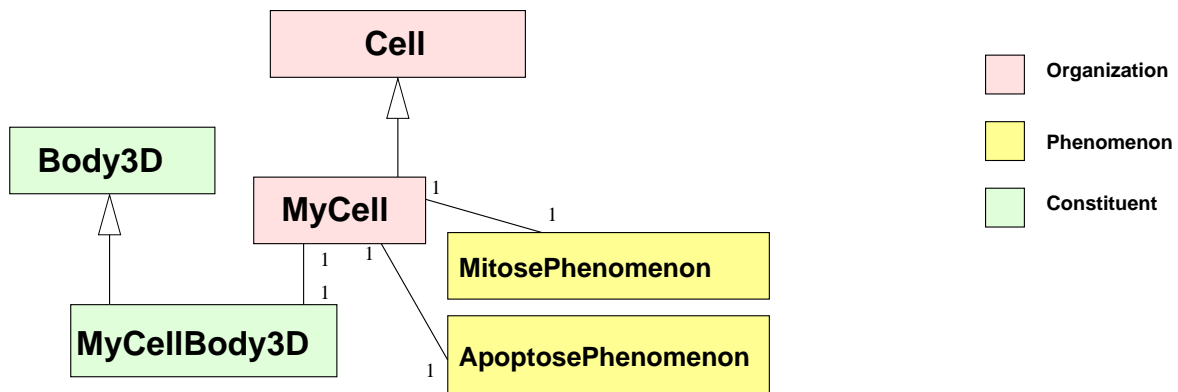


FIG. 7.9: L'organisation `SourceEnergie` est composée de 2 réactions; la première produit du glucose et la seconde permet sa libération dans le milieu. La source d'énergie est donc couplée avec l'organisation chimique. Les constituants `Compartment` ne sont pas représentés.

FIG. 7.10: Diagramme UML de la classe `MyCell`.

déclenchement de la division n'intervient que si les espèces « mitoseActivateur » et « énergie » dépassent un certain seuil. Enfin, si l'espèce « régulateur » dépasse la concentration critique, la mitose est inhibée.

Le phénomène d'apoptose (mort cellulaire programmée) est quant à lui subordonné au niveau d'énergie de la cellule. Si la cellule vient à manquer de glucose pendant une période trop importante, la concentration d'énergie peut descendre au-dessous d'un certain seuil. Dans ce cas, l'interaction apoptose déclenche la destruction de la cellule.

Bien sûr, il est possible d'étoffer le phénomène en le liant à d'autres paramètres pour établir une mort cellulaire plus « programmée ». De même, le phénomène de la mitose pourrait être amélioré. Toutefois, ces deux phénomènes sont suffisants pour décrire la sous-unité de base qu'est notre cellule autonome.

7.2.3.3 L'organisation `AutopoieticOrganisation`

Nous définissons le système autopoïétique par son organisation⁴ : `AutopoieticOrganisation`. L'organisation `AutopoieticOrganisation` est d'abord composée de la première cellule, puis de toutes les nouvelles cellules issues de la division cellulaire. Ensuite, deux nouveaux phénomènes ont été ajoutés pour assurer le maintien de l'arrangement spatial des cellules (voir figure 7.12).

Le premier phénomène `OrientationPhenomenon` instancie des interactions binaires de type `Orientation` entre les corps des cellules. Chaque corps de cellule possède une pastille rouge. L'interaction agit sur l'orientation des cellules pour faire en sorte de repousser cette pastille. Le phénomène a alors pour action d'orienter toutes les pastilles des cellules vers l'extérieur du système. Le second phénomène `AgitationPhenomenon` instancie des interactions binaires entre chaque corps de cellule et le milieu. Cette nouvelle interaction fait migrer le corps d'une cellule dans la direction opposée à la pastille.

Ainsi la combinaison des deux phénomènes a pour effet de rassembler toutes les cellules du système. Nous reproduisons là ce que pourrait être l'action de pseudopodes de la cellule réagissant à un gradient de concentration de médiateur produit par l'ensemble des cellules. Ces phénomènes nous montrent au passage qu'il est possible de modéliser directement des interactions au niveau biologique et que nous ne sommes pas obligé de modéliser uniquement des phénomènes physico-chimiques (ce qui reviendrait à adopter une vision mécaniste).

⁴Rappelons que dans le cadre de notre modèle générique, un système autonome se définit complètement par une organisation et par l'ensemble des constituants auxquels elle est associée. Du point de vue de

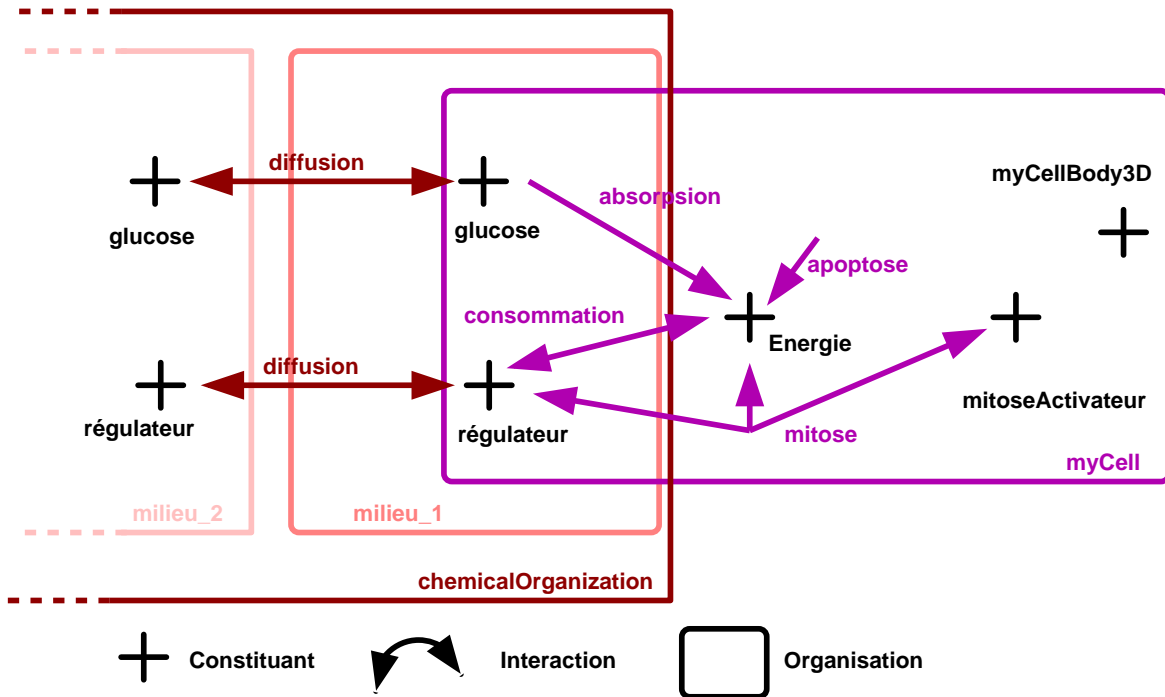


FIG. 7.11: Principe de l'organisation MyCell. Les constituants Compartment ne sont pas représentés.

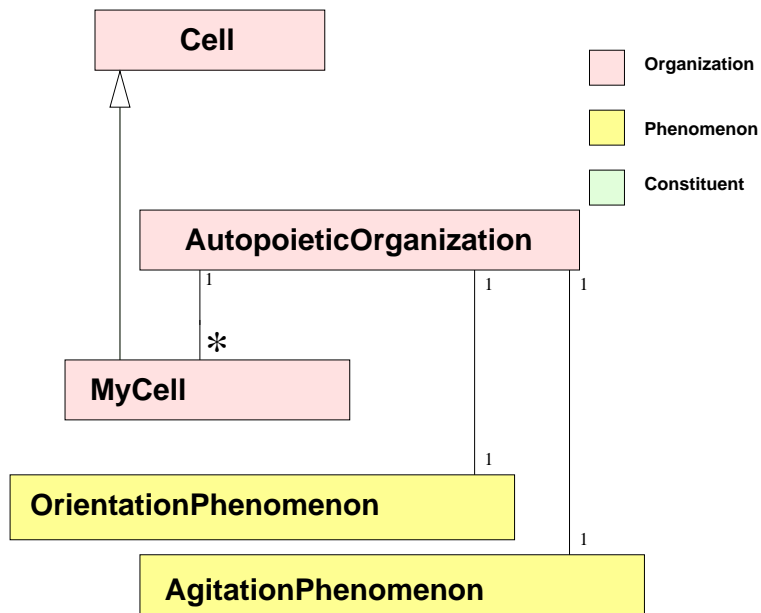


FIG. 7.12: Diagramme UML de la classe AutopoieticOrganization.

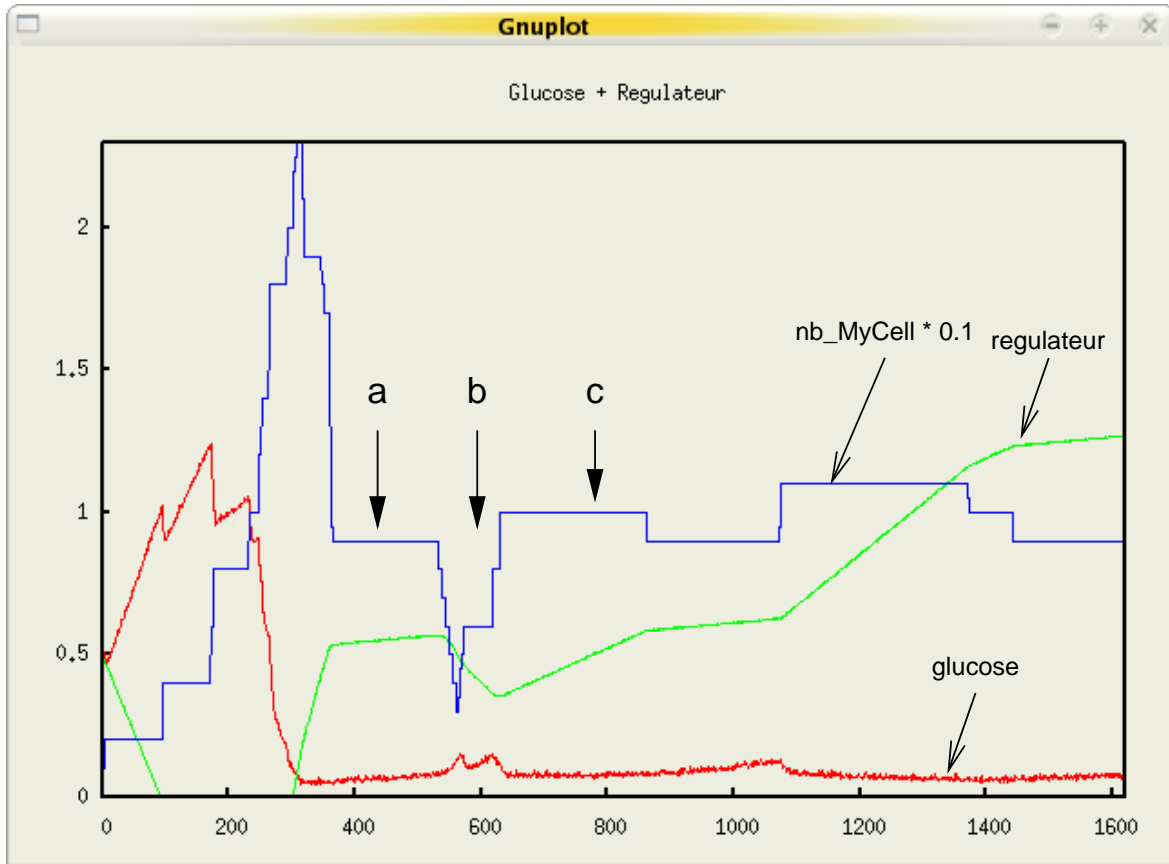


FIG. 7.13: Au départ, le nombre de cellules croît jusqu'au manque d'énergie. Il se stabilise alors à 9 cellule (a). Ensuite, lorsqu'on perturbe le système en détruisant un certain nombre de cellules (b), on observe le rétablissement de l'équilibre (c). Enfin, la concentration de régulateur atteint un seuil qui permet d'inhiber la croissance du système lorsqu'il a atteint la taille souhaitée.

7.2.4 Expérimentation

À l'initialisation, nous créons un environnement (`ChemicalOrganization` et `SourceEnergie`) puis une organisation `AutopoieticOrganization`. Le graphique de la figure 7.13 montre l'évolution de la population de cellules au cours du temps.

On observe alors un régime transitoire puis un régime stable. En effet le système cherche à maintenir une population de 9 ou 10 cellules. Lorsque l'on perturbe le système en le détruisant partiellement, il se régénère pour atteindre un nouvel état stable.

L'homogénéité spatiale du système est assurée par les interactions d'orientation et d'agitation. Finalement on observe un amas de cellules, centripète de taille constante. La figure 7.14 montre une capture d'écran du système.

L'amas cellulaire peut être vu comme un *pattern* dynamique et émergent. Observons de plus, que le système à une durée de vie bien plus longue que les cellules qui se régénèrent au cours de la simulation. Le système est alors bien traversé par un flux continu d'énergie et de matière.

l'implémentation, il n'y a pas de différence entre une organisation et un système autonome.

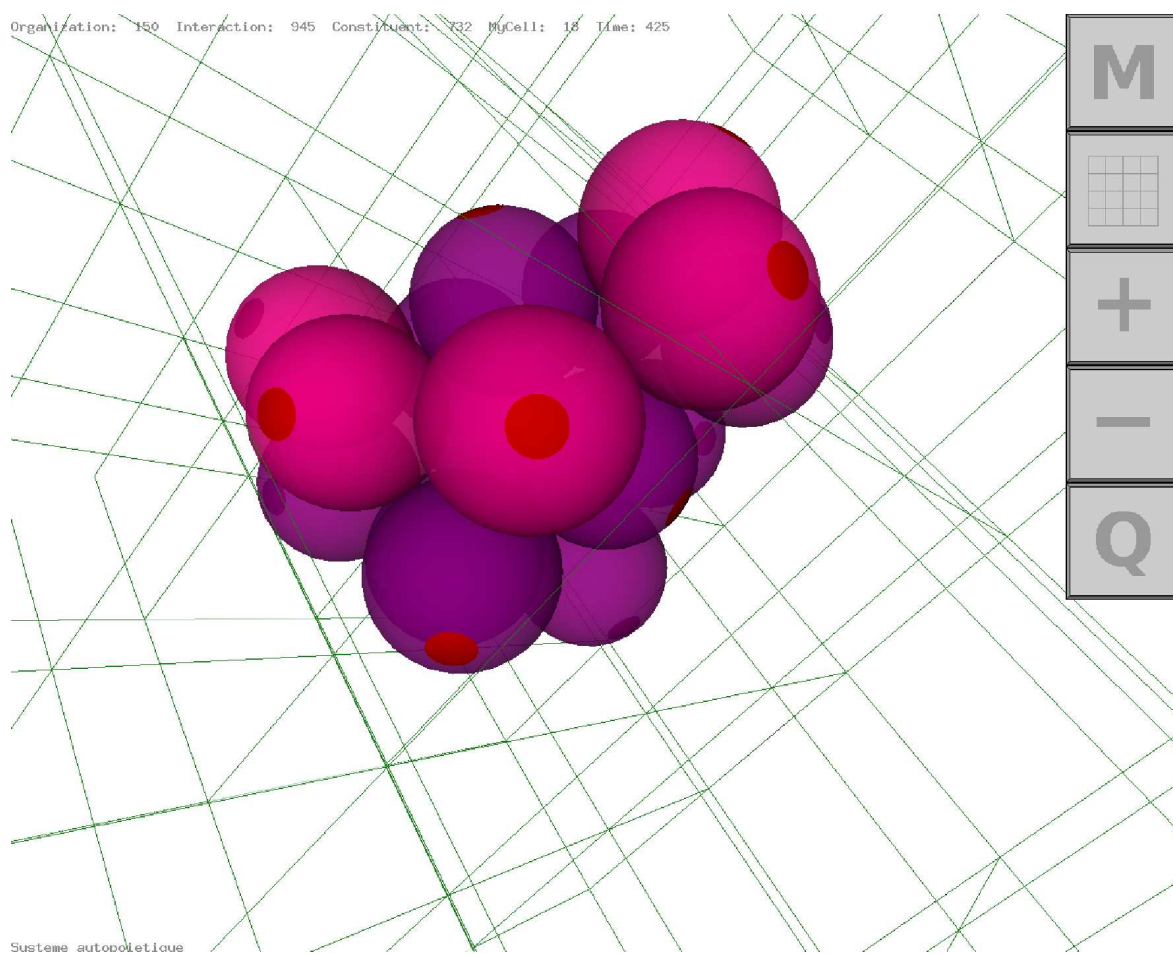


FIG. 7.14: Amas cellulaire composant le système autopoïétique.

7.2.5 Discussion

Notre modèle minimaliste de système autopoïétique n'apporte évidemment pas grand chose ni à l'étude de l'auto-organisation et ni à celle de l'autopoïèse. Cependant, il permet de montrer la simplicité avec laquelle il est possible de définir de tels systèmes en utilisant notre cadre de modélisation.

Durant l'exécution du modèle, l'organisation `AutopoieticOrganization` est naturellement composée de l'ensemble des sous-unités qui composent le système (les cellules). De plus, il est possible de spécifier les phénomènes au niveau de description auquel ils nous semblent les plus pertinents. Notons que tous les éléments de la structure qui appartiennent aux organisations de premier ordre `MyCell` appartiennent également à l'organisation supérieure `AutopoieticOrganization`. De ce fait, les compartiments et espèces chimiques du milieu couplés aux cellules appartiennent également au système autopoïétique, si bien qu'il se définit par toute une région de l'espace. Pour reprendre la définition : il est constitué en tant qu'unité concrète de l'espace où il existe.

Ensuite, nous constatons que le système se régénère continuellement. Le système peut se déplacer dans le milieu en fonction des positions de ses sous-unités⁵. La transformation du « glucose » en « régulateur » constitue un flux de matière et d'énergie qui traverse le système lui permettant de se maintenir. De plus, la mort cellulaire et les perturbations du niveau d'énergie font que le système dure plus longtemps que n'importe laquelle des sous-unités qui y participe.

Le modèle permet donc de modéliser l'évolution d'un système autopoïétique à partir de la modélisation de lois physiques, biologiques et de toute autre nature jugée intéressante par le modélisateur. Nous travaillons à l'ajout d'interactions géométriques permettant une auto-organisation spatiale du système. Par exemple, des interactions entre le volume des compartiments et celui du corps des cellules pourraient notamment limiter les ressources en énergie des cellules prisonnières au centre de l'amas. En cela le maillage adaptatif présenté à la section précédente pourrait se révéler utile.

Ajoutons que la possibilité de faire intervenir la médiation des sens pour enrichir le modèle au cours de l'expérimentation présente une perspective vraiment intéressante pour l'étude de systèmes autopoïétiques, tant il est difficile de prévoir *a priori* les valeurs des paramètres des différents phénomènes de régulation.

Enfin, nous avons introduit deux phénomènes très particuliers dans nos simulations : l'apoptose et la mitose. Leur mise en œuvre n'a suscité aucun problème technique. En effet les méthodes de clonage d'objet étant déjà définies dans la `libBio`, la copie de l'objet de type `MyCell` est immédiate, et il en va de même pour la destruction des organisations.

Cependant, telles que nous avons conçu ces interactions, le constructeur et le destructeur de l'organisation `MyCell` sont directement appelés par un objet de type `Interaction`. Or jusqu'ici le champ d'action des interactions était circonscrit à la manipulation de la structure (plus précisément à l'ensemble des constituants qui lui sont associés). En autorisant une interaction à créer ou à supprimer une cellule, on la laisse intervenir dans l'agencement des organisations, se faisant elle se substitue au rôle du modélisateur. Jusqu'ici seules les organisations étaient habilitées à jouer ce rôle.

Le problème de la mitose soulève une critique souvent émise à l'égard des travaux de Varela sur la modélisation du vivant. La reproduction y est en effet souvent reléguée au second plan, car elle n'est pas pour lui, considérée comme une propriété nécessaire à la vie. Ainsi le modèle proposé dans cette thèse ne nous dit rien sur la façon de modéliser le phénomène de la

⁵Le système a d'ailleurs tendance à se rapprocher de la source d'énergie.

reproduction. Parmi la variété de solutions techniques à notre disposition, nous avons choisi la plus immédiate, il faudrait toutefois mener une réflexion plus approfondie sur ce sujet.

En fait la question de la reproduction soulève une autre question plus vaste. La plupart des travaux modélisant les interactions ne modélisent pas la dynamique de création et de destruction des interactions. Le réseau des interactions est fixé par le modélisateur et la dynamique du système est réalisée par les interactions. Le modèle que propose cette thèse transforme le rôle du modélisateur qui doit décrire les modes d'évolution du réseau en plus de la dynamique réalisée par les interactions. En cela, la notion de système autonome nous a aidés. Nous avons alors considéré la création des systèmes autonomes comme étant un acte subjectif de modélisation (voir section 4.3). Ensuite chaque système est en charge du maintien de sa topologie. La question posée par la mitose nous amène à nous demander comment le modèle peut automatiser l'émergence d'un nouveau système et ainsi se substituer au rôle du modélisateur. Cette thèse est une étape qui vise à modéliser et organiser les conditions d'émergence des nouvelles interactions, l'étape suivante est de modéliser les conditions d'émergence des systèmes autonomes. Nous laissons donc la question ouverte. Ce qui ne nous limite pas pour l'instant puisque la mitose ne nous pose aucun problème d'ordre technique.

Concluons ce chapitre en indiquant que l'étude de l'autopoïèse pourrait être une perspective d'utilisation et de développement du modèle générique. La propriété d'autonomie étant naturellement conférée aux systèmes définis par les organisations, l'ajout de mécanismes de production et de maintien de l'homéostasie suffit à recréer l'autopoïèse dans nos univers virtuels.

Conclusions

Le virtuoscope désigne un laboratoire virtuel pour l'étude des systèmes complexes, qui s'appuie sur les concepts, les modèles et les outils de la réalité virtuelle. Il permet un nouveau mode d'investigation nommé expérimentation *in virtuo*.

Cette thèse s'inscrit dans le cadre de l'application du virtuoscope à l'étude de systèmes physiologiques humains, et plus généralement à la biologie. Nous dressons alors le bilan du travail réalisé, puis, nous exposons quelques perspectives offertes par ce travail.

Bilan

Dans ce mémoire, nous avons présenté un modèle générique permettant de fédérer et de factoriser les différentes applications réalisées dans le cadre de l'expérimentation *in virtuo*, pour l'étude du vivant. Ce modèle repose d'une part, sur le principe original des interactions réifiées en objets autonomes ordonnancés de manière asynchrone et chaotique, et d'autre part, sur une organisation des modèles en systèmes autonomes en couplage structurel. Pour présenter ce travail, nous avons choisi le cheminement suivant : état de l'art, modèle et application.

Présentation du contexte interdisciplinaire. Tout d'abord, nous avons proposé un état de l'art en trois parties pour positionner ce travail ancré dans la recherche interdisciplinaire. Le premier chapitre a fait état des évolutions de la biologie et de l'émergence de nouveaux paradigmes pour la prise en compte de la complexité. Le deuxième chapitre nous a permis de faire un tour d'horizon des méthodes informatiques permettant de traiter la complexité biologique. Nous nous sommes plus particulièrement intéressés aux calculs *in silico* qui permettent l'analyse *a posteriori*. Puis, le troisième chapitre, à travers les notions de cognition et d'expérience, nous a permis de comprendre l'intérêt de la réalité virtuelle pour appréhender les systèmes biologiques que nous modélisons. Ainsi, ce travail se situe au carrefour d'au moins trois disciplines : l'étude des systèmes complexes en biologie, les calculs *in silico* et la réalité virtuelle. Ces trois domaines forment alors un système de trois « équations » auquel nous souhaitons fournir une solution qui se matérialise par un modèle générique.

Proposition d'un cadre de modélisation. Nous avons décrit le modèle générique, puis nous avons présenté la `libBio`, la « boîte à outils » utilisant ce modèle. Le modèle se compose des 5 classes de base : `Interaction`, `Phénomène`, `Organisation`, `Structure` et `Constituant`. Les objets de type `Interaction` sont des objets actifs qui doivent

être ordonnancés de manière chaotique et asynchrone. La méthode proposée permet alors la simulation de systèmes multi-échelles (de temps et d'espace), multi-niveaux (d'organisation) et multi-modèles (de natures différentes) ; ce qui répond à un réel besoin de la modélisation en biologie.

Ensuite, nous avons dérivé le modèle générique pour illustrer la prise en compte des phénomènes de natures chimiques, biologiques (cellule) ou mécaniques ; montrant ainsi les aspects maintenables et modulaires du code informatique obtenu.

Enfin nous avons précisé la place du modèle de l'utilisateur pour nous assurer que le modèle générique était intrinsèquement adapté aux contraintes de la réalité virtuelle.

Application de la méthode proposée. Le meilleur moyen de valider ou tout au moins de tester une méthode de modélisation, c'est bien sûr de l'appliquer à des cas concrets d'utilisation. Le modèle de réaction allergique réalisé en collaboration avec une équipe de biologistes a permis une réelle confrontation avec les réalités d'une certaine recherche en biologie expérimentale. L'application alors réalisée a montré que la méthode autorisait le travail interdisciplinaire. Elle offre de plus, la possibilité essentielle de traiter la modélisation des systèmes de la biologie intégrative de façon modulaire et incrémentale. Puis, nous avons montré comment il devenait possible d'entrevoir la modélisation des phénomènes de réaction-diffusion sous un angle différent, et plus généralement, comment l'autonomie des modèles leur permet de structurer eux-mêmes leur environnement. Pour finir, nous avons modélisé un système autopoïétique minimal montrant qu'il était possible d'orienter l'utilisation du modèle générique vers des travaux plus théoriques.

Ce travail représente donc une étape dans le projet du virtuoscope et dans l'établissement du mode d'expérimentation *in virtuo*. Le chapitre 3 a largement développé les avantages d'un tel moyen d'investigation. Cependant, le cadre générique de modélisation proposé enrichit le domaine d'un certain nombre de points forts. Nous rappelons alors quelques-unes des spécificités du modèle générique qui peuvent présenter un intérêt en dehors du cadre du virtuoscope :

- Le modèle générique est intrinsèquement orienté multi-modèles, il fournit une expressivité très importante. Il permet ainsi le travail interdisciplinaire offrant un outil pour la biologie expérimentale et théorique.
- Une solution générique pour gérer la dynamique de l'organisation des modèles est proposée, là où les rares modèles d'organisation proposés dans la littérature sont statiques.
- Le modèle générique n'utilise que des aspects classiques de la programmation orientée objets. Il peut ainsi être implémenté aux moyens de l'ensemble des langages orientés objets. Il a pour objectif la production de code structuré, maintenable et surtout réutilisable par des utilisateurs ne possédant que les bases de la programmation orientée objets.
- L'inscription de ce travail dans le contexte d'un laboratoire de réalité virtuelle a autorisé la conception d'un outil efficace pour la modélisation de phénomènes biologiques dans les trois dimensions de l'espace.
- La méthode proposée oriente la modélisation vers une organisation des modèles adaptée aux différents paradigmes de la biologie théorique, systémique ou intégrative. Elle est

alors propice à faire le lien entre biologie expérimentale et biologie théorique.

Les travaux réalisés au cours de la thèse ont donné lieu à publications dans le domaine de l'informatique : Desmeulles et al. [2005b,c, 2006]. Ils ont également suscité l'intérêt de la communauté des dermato-allergologues à travers le modèle de peau : Desmeulles et al. [2003a,c,b, 2005d,a]. Finalement ces travaux venant enrichir le virtuoscope, ouvrent un certain nombre de pistes de recherche.

Perspectives

Nous classons les perspectives ouvertes par ce travail en trois catégories. Tout d'abord, nous présentons les projets d'évolution du modèle générique. Ensuite, la seconde catégorie fait référence à un certain nombre de voies pour des travaux visant à améliorer l'accessibilité au virtuoscope. Enfin, la troisième représente le travail applicatif qu'il est désormais possible de mettre en œuvre.

Évolution du modèle

Le modèle générique a été présenté comme une solution d'implémentation du virtuoscope pour l'étude de systèmes biologiques. Il a notamment permis d'intégrer au sein d'une bibliothèque les outils développés au CERV dans ce domaine. En cela, il représente une étape nécessaire au développement de la méthode *in virtuo*. Il convient maintenant d'envisager son amélioration par les travaux à venir.

La taille des simulations est aujourd'hui la principale limite à laquelle doivent faire face les applications du modèle générique. La distribution des calculs est envisagée pour pallier ce problème. Il paraît alors préférable d'intégrer rapidement au modèle générique, les contraintes inhérentes à la répartition des modèles, avant de lui appliquer d'autres évolutions. La question de l'ordonnancement des interactions (traitée à la section 4.2) est cruciale à ce niveau. Actuellement, l'étude de la distribution des modèles *in virtuo* fait l'objet d'une nouvelle thèse⁶.

Ensuite, comme nous l'avons indiqué à la section 5.3.3, nous envisageons de doter les **Phénomène** de la possibilité de réaliser des interactions ponctuelles. La réponse à cette question est liée à l'ordonnancement, et par conséquent, elle ne pourra être abordée qu'après la question de la distribution des modèles.

Enfin, la question de l'émergence automatisée de nouveaux systèmes autonomes au sein des univers virtuels, posée au dernier chapitre, pourra être abordée. La réponse fournira notamment un modèle de reproduction.

La construction du modèle générique permet de répondre aux questions relatives à la méthode de construction des modèles numériques. Les évolutions du modèle générique ne sont bien sûr pas les seuls aspects de la démarche *in virtuo* qui doivent être développés. Ainsi, le développement des interfaces et de l'accessibilité au laboratoire virtuel par des utilisateurs non informaticiens sont des étapes cruciales qui devront naturellement être abordées.

⁶Thèse en court de réalisation par Mickaël Bourhis au CERV. Soutenance prévue en 2008.

Interfaces et interdisciplinarité

Il devient aujourd'hui communément admis qu'il n'y a plus d'intérêt à chercher l'« équation qui régit le monde ». De la même manière, il est assez illusoire de chercher « LA » théorie unificatrice expliquant tout. L'expérimentation *in virtuo* est un moyen de mettre en œuvre simultanément des modèles issus de cadres théoriques variés. Nous avons placé ce point de vue au centre de nos préoccupations car il est fondamental en biologie, science mal définie théoriquement. Le modèle générique a alors apporté une solution logicielle pour intégrer ces modèles dans une application de réalité virtuelle. Remarquons alors que le champ d'application de la méthode proposée ne se limite pas à l'étude des systèmes biologiques.

L'aspect multi-modèles a permis de construire des systèmes en collaboration avec différents experts. L'expérimentation *in virtuo* autorise réellement un travail interdisciplinaire. Cependant, sans la présence d'interfaces adéquates, seul un informaticien peut construire les modèles. L'étude des interfaces doit donc s'ancrer dans cette interdisciplinarité. Il faudra alors mener un travail spécifique pour proposer les interfaces permettant aux utilisateurs non informaticiens d'intervenir eux-mêmes sur le modèle *a priori*, et surtout leur permettre de réaliser l'expérience des modèles par la triple médiation des sens, de l'action et de l'esprit.

Le domaine de la biologie souffre encore plus que la plupart des autres domaines scientifiques du cloisonnement de ses sous-disciplines et de l'ultra spécialisation de ses chercheurs. Offrir un outil de modélisation pour le travail interdisciplinaire et réunir différents chercheurs autour d'une maquette virtuelle à laquelle chacun peut ajouter son savoir et dont chacun peut faire l'expérience pour en améliorer sa connaissance, est probablement la perspective la plus enthousiasmante pour la biologie intégrative.

Applications

Bien entendu, un certain nombre de perspectives applicatives succèdent à l'établissement d'un modèle générique. De plus, les travaux menés au CERV pour l'étude de systèmes biologiques l'ont tous été en collaboration avec une équipe de recherche du domaine (hématologues, cancérologues, immunologues, dermatologues...). L'avenir de la méthode *in virtuo*, passera pas la réalisation de nouvelles applications et par le lancement de nouvelles collaborations, car c'est la confrontation aux problèmes concrets qui permet de se poser les bonnes questions. Toutefois, la nouveauté de la notion d'interaction réifiée fait qu'il est encore difficile de prédire précisément vers quels types de modèles applicatifs nous nous dirigerons. Il est cependant possible de classer les différentes voies à suivre.

En premier lieu, le dernier chapitre a présenté deux applications en cours de développement. Le maillage adaptatif doit être testé et intégré dans la `libBio`. La piste de l'autopoïèse⁷ peut également être poursuivie pour étoffer l'exemple proposé.

Deuxièmement, nous pouvons nous demander s'il ne serait pas intéressant de modéliser des exemples de systèmes bien connus afin de valider la méthode. C'est en effet la méthode qui est en général préconisée par la recherche *in silico*. En fait, nous considérons trois niveaux hiérarchiques dans les modèles : le modèle générique, la dérivation de celui-ci en plusieurs classes de phénomènes particuliers et les modèles biologiques dérivant de l'un ou l'autre des deux niveaux supérieurs. Nous pensons alors que la modélisation de phénomènes bien connus permet au mieux de valider les deux niveaux inférieurs : les modèles biologiques et les classes de phénomènes, mais ne permet pas de valider le modèle générique. Dans ce cas, ce type d'application ne trouvera son utilité que pour valider de nouveaux phénomènes qui seront

⁷Voir la thèse de Kristen Manac'h, réalisée au CERV. Soutenance prévue fin 2009.

ajoutés à la `libBio`.

Troisièmement, l'outil *in virtuo* est conçu pour donner du sens aux systèmes complexes étudiés. Plutôt que de s'intéresser à la modélisation de systèmes connus, il paraît plus intéressant de se diriger vers des domaines en quête de compréhension. Le domaine de la dermato-allergologie auquel s'associe l'application présentée dans cette thèse répond à ce critère. Il faut alors envisager de s'attaquer à la modélisation de systèmes partiellement compris appartenant à de tels domaines.

Finalement les perspectives applicatives sont très variées, l'existence de la `libBio` constitue un outil réutilisable qui évoluera et s'enrichira à chaque nouvelle application. Au fil de l'évolution de la `libBio`, le temps de développement des applications tend à diminuer. Il apparaît qu'à terme, le temps de développement deviendra négligeable par rapport au temps de rassemblement et de production des données, d'une part, et au temps de construction et d'expérimentation des modèles biologiques, d'autre part.

Annexes

Annexe A

Mast.xml

```
<?xml version="1.0" ?>
<!-- MAST -->

<sbml xmlns="http://www.sbml.org/sbml/level2" version="1" level="2">

<model id="mast">

<listOfCompartments>
<compartment id="milieu" size="1"/>
<compartment id="membrane" size="1"/>
<compartment id="cytoplasme" size="1"/>
</listOfCompartments>

<listOfSpecies>
<!-- Les recepteurs en 10^-6-->
<species id="FcERI" initialAmount="0.000000000025" compartment="membrane"/>
<species id="FcERI_Allergene" initialAmount="0.0" compartment="membrane"/>

<species id="NK" initialAmount="0.000000000001" compartment="membrane"/>
<species id="NK_Substance_P" initialAmount="0" compartment="membrane"/>

<species id="TnfInternePre" initialAmount="0.0000000117" compartment="cytoplasme"/>
<species id="PreTnfInterneNeoA" initialAmount="0.0000000351" compartment="cytoplasme"/>
<species id="PreTnfInterneNeoB" initialAmount="0" compartment="cytoplasme"/>
<species id="PreTnfInterneNeoC" initialAmount="0" compartment="cytoplasme"/>
<species id="TnfInterneNeo" initialAmount="0" compartment="cytoplasme"/>

<species id="Histamine" initialAmount="0.000000036" compartment="cytoplasme"/>
<species id="ActivateurH" initialAmount="0" compartment="cytoplasme"/>
<species id="InhibiteurH" initialAmount="0" compartment="cytoplasme"/>
<species id="ActivateurSP" initialAmount="0" compartment="cytoplasme"/>
<species id="InhibiteurSP" initialAmount="0" compartment="cytoplasme"/>

<species id="Tryptase" initialAmount="0.0000000026" compartment="cytoplasme"/>
<species id="ActivateurTryptase" initialAmount="0" compartment="cytoplasme"/>
<species id="InhibiteurTryptase" initialAmount="0" compartment="cytoplasme"/>

<species id="TNFalpha_milieu" initialAmount="0" compartment="milieu"/>
<species id="Allergene_milieu" initialAmount="0" compartment="milieu"/>
<species id="Histamine_milieu" initialAmount="0" compartment="milieu"/>
<species id="Tryptase_milieu" initialAmount="0" compartment="milieu"/>
<species id="Substance_P_milieu" initialAmount="0" compartment="milieu"/>
</listOfSpecies>
```

```
<listOfReactions>
  <!-- production FcERI_Allergene -->
  <reaction id="FcERI_Allergene creation" reversible="false">
    <listOfReactants>
      <speciesReference species="FcERI" compartment="membrane" stoichiometry="1"/>
      <speciesReference species="Allergene_milieu" compartment="milieu" stoichiometry="1"/>
    </listOfReactants>
    <listOfProducts>
      <speciesReference species="FcERI_Allergene" compartment="membrane" stoichiometry="1"/>
    </listOfProducts>
    <kineticLaw>
      <math xmlns="http://www.w3.org/1998/Math/MathML">
        <apply>
          <times/>
          <ci>ka</ci>
          <ci>FcERI</ci>
          <ci>Allergene_milieu</ci>
        </apply>
      </math>
      <listOfParameters>
        <parameter id="ka" value="56e9"/>
        <!-- parameter id="ka" value="70000"-->
      </listOfParameters>
    </kineticLaw>
  </reaction>

  <!-- production NK_Substance P -->
  <reaction id="NK_Substance P creation" reversible="false">
    <listOfReactants>
      <speciesReference species="NK" compartment="membrane" stoichiometry="1"/>
      <speciesReference species="Substance_P_milieu" compartment="milieu" stoichiometry="1"/>
    </listOfReactants>
    <listOfProducts>
      <speciesReference species="NK_Substance_P" compartment="membrane" stoichiometry="1"/>
    </listOfProducts>
    <kineticLaw>
      <math xmlns="http://www.w3.org/1998/Math/MathML">
        <apply>
          <times/>
          <ci>ka</ci>
          <ci>NK</ci>
          <ci>Substance_P_milieu</ci>
        </apply>
      </math>
      <listOfParameters>
        <parameter id="ka" value="4e9"/>
        <!--parameter id="ka" value="4e10"-->
      </listOfParameters>
    </kineticLaw>
  </reaction>
</listOfReactions>
```



```

<!-- ||| HISTAMINE ||| -->

<!-- ===== relargage Histamine par FcERI -->
<reaction id="ActivationH" reversible="false">
  <listOfModifiers>
    <modifierSpeciesReference species="FcERI_Allergene"
                             compartment="membrane" stoichiometry="1"/>
  </listOfModifiers>
  <listOfProducts>
    <speciesReference species="ActivateurH" compartment="cytoplasme" stoichiometry="1"/>
  </listOfProducts>
  <kineticLaw>
    <math xmlns="http://www.w3.org/1998/Math/MathML">
      <apply>
        <times/>
        <ci>k1</ci>
        <ci>FcERI_Allergene</ci>
        <apply>
          <plus/>
          <ci>un</ci>
        </apply>
        <apply>
          <times/>
          <ci>moinsUn</ci>
          <ci>ActivateurH</ci>
        </apply>
      </apply>
    </math>
    <listOfParameters>
      <parameter id="k1" value="1000000"/>
      <parameter id="un" value="1"/>
      <parameter id="moinsUn" value="-1"/>
    </listOfParameters>
  </kineticLaw>
</reaction>

<reaction id="InhibitionH" reversible="false">
  <listOfModifiers>
    <modifierSpeciesReference species="ActivateurH"
                             compartment="cytoplasme" stoichiometry="1"/>
  </listOfModifiers>
  <listOfProducts>
    <speciesReference species="InhibiteurH" compartment="cytoplasme" stoichiometry="1"/>
  </listOfProducts>
  <kineticLaw>
    <math xmlns="http://www.w3.org/1998/Math/MathML">
      <apply>
        <times/>
        <ci>k2</ci>
        <ci>ActivateurH</ci>
        <apply>
          <plus/>
          <ci>un</ci>
        </apply>
        <apply>
          <times/>
          <ci>moinsUn</ci>
          <ci>InhibiteurH</ci>
        </apply>
      </apply>
    </math>
    <listOfParameters>
      <parameter id="k2" value="0.00610"/>
      <parameter id="un" value="1"/>
      <parameter id="moinsUn" value="-1"/>
    </listOfParameters>
  </kineticLaw>
</reaction>

```

```

<reaction id="liberationH" reversible="false">
  <listOfReactants>
    <speciesReference species="Histamine" compartment="cytoplasme" stoichiometry="1"/>
  </listOfReactants>
  <listOfModifiers>
    <modifierSpeciesReference species="ActivateurH"/>
    <modifierSpeciesReference species="InhibiteurH"/>
  </listOfModifiers>
  <listOfProducts>
    <speciesReference species="Histamine_milieu" compartment="milieu" stoichiometry="1"/>
  </listOfProducts>
  <kineticLaw>
    <math xmlns="http://www.w3.org/1998/Math/MathML">
      <apply>
        <times/>
        <apply>
          <plus/>
          <ci>ActivateurH</ci>
          <apply>
            <times/>
            <ci>moinsUn</ci>
            <ci>InhibiteurH</ci>
          </apply>
        </apply>
        <ci>k3</ci>
        <ci>Histamine</ci>
      </apply>
    </math>
    <listOfParameters>
      <parameter id="k3" value="0.17"/>
      <parameter id="un" value="1"/>
      <parameter id="moinsUn" value="-1"/>
    </listOfParameters>
  </kineticLaw>
</reaction>

<!-- ===== relargage Histamine par Substance P -->
<reaction id="ActivationH" reversible="false">
  <listOfModifiers>
    <modifierSpeciesReference species="NK_Substance_P" compartment="membrane"
      stoichiometry="1"/>
  </listOfModifiers>
  <listOfProducts>
    <speciesReference species="ActivateurSP" compartment="cytoplasme" stoichiometry="1"/>
  </listOfProducts>
  <kineticLaw>
    <math xmlns="http://www.w3.org/1998/Math/MathML">
      <apply>
        <times/>
        <ci>k1</ci>
        <ci>NK_Substance_P</ci>
        <apply>
          <plus/>
          <ci>un</ci>
          <apply>
            <times/>
            <ci>moinsUn</ci>
            <ci>ActivateurSP</ci>
          </apply>
        </apply>
      </math>
    <listOfParameters>
      <parameter id="k1" value="50000000"/>
      <parameter id="un" value="1"/>
      <parameter id="moinsUn" value="-1"/>
    </listOfParameters>
  </kineticLaw>
</reaction>

```

```

<reaction id="InhibitionH" reversible="false">
  <listOfModifiers>
    <modifierSpeciesReference species="ActivateurSP" compartment="cytoplasme"
                              stoichiometry="1"/>
  </listOfModifiers>
  <listOfProducts>
    <speciesReference species="InhibiteurSP" compartment="cytoplasme" stoichiometry="1"/>
  </listOfProducts>
  <kineticLaw>
    <math xmlns="http://www.w3.org/1998/Math/MathML">
      <apply>
        <times/>
        <ci>k2</ci>
        <ci>ActivateurSP</ci>
        <apply>
          <plus/>
          <ci>un</ci>
          <apply>
            <times/>
            <ci>moinsUn</ci>
            <ci>InhibiteurSP</ci>
          </apply>
        </apply>
      </apply>
    </math>
    <listOfParameters>
      <parameter id="k2" value="0.2270"/>
      <parameter id="un" value="1"/>
      <parameter id="moinsUn" value="-1"/>
    </listOfParameters>
  </kineticLaw>
</reaction>

<reaction id="liberationH" reversible="false">
  <listOfReactants>
    <speciesReference species="Histamine" compartment="cytoplasme" stoichiometry="1"/>
  </listOfReactants>
  <listOfModifiers>
    <modifierSpeciesReference species="ActivateurSP"/>
    <modifierSpeciesReference species="InhibiteurSP"/>
  </listOfModifiers>
  <listOfProducts>
    <speciesReference species="Histamine_milieu" compartment="milieu" stoichiometry="1"/>
  </listOfProducts>
  <kineticLaw>
    <math xmlns="http://www.w3.org/1998/Math/MathML">
      <apply>
        <times/>
        <apply>
          <plus/>
          <ci>ActivateurSP</ci>
          <apply>
            <times/>
            <ci>moinsUn</ci>
            <ci>InhibiteurSP</ci>
          </apply>
        </apply>
        <ci>k3</ci>
        <ci>Histamine</ci>
      </apply>
    </math>
    <listOfParameters>
      <parameter id="k3" value="0.17"/>
      <parameter id="un" value="1"/>
      <parameter id="moinsUn" value="-1"/>
    </listOfParameters>
  </kineticLaw>
</reaction>

```

```

<!-- ||| TRYPTASE ||| -->

<!-- relargage Tryptase -->
<reaction id="ActivationTryptase" reversible="false">
  <listOfModifiers>
    <modifierSpeciesReference species="FcERI_Allergene" compartment="membrane"
                              stoichiometry="1"/>
  </listOfModifiers>
  <listOfProducts>
    <speciesReference species="ActivateurTryptase" compartment="cytoplasme"
                      stoichiometry="1"/>
  </listOfProducts>
  <kineticLaw>
    <math xmlns="http://www.w3.org/1998/Math/MathML">
      <apply>
        <times/>
        <ci>k1</ci>
        <ci>FcERI_Allergene</ci>
        <apply>
          <plus/>
          <ci>un</ci>
          <apply>
            <times/>
            <ci>moinsUn</ci>
            <ci>ActivateurTryptase</ci>
          </apply>
        </apply>
      </math>
    </kineticLaw>
  </reaction>

<reaction id="InhibitionTryptase" reversible="false">
  <listOfModifiers>
    <modifierSpeciesReference species="ActivateurTryptase" compartment="cytoplasme"
                              stoichiometry="1"/>
  </listOfModifiers>
  <listOfProducts>
    <speciesReference species="InhibiteurTryptase" compartment="cytoplasme" stoichiometry="1"/>
  </listOfProducts>
  <kineticLaw>
    <math xmlns="http://www.w3.org/1998/Math/MathML">
      <apply>
        <times/>
        <ci>k2</ci>
        <ci>ActivateurTryptase</ci>
        <apply>
          <plus/>
          <ci>un</ci>
          <apply>
            <times/>
            <ci>moinsUn</ci>
            <ci>InhibiteurTryptase</ci>
          </apply>
        </apply>
      </math>
    </kineticLaw>
  </reaction>

```

```

<reaction id="liberationTryptase" reversible="false">
  <listOfReactants>
    <speciesReference species="Tryptase" compartment="cytoplasme" stoichiometry="1"/>
  </listOfReactants>
  <listOfModifiers>
    <modifierSpeciesReference species="ActivateurTryptase"/>
    <modifierSpeciesReference species="InhibiteurTryptase"/>
  </listOfModifiers>
  <listOfProducts>
    <speciesReference species="Tryptase_milieu" compartment="milieu" stoichiometry="1"/>
  </listOfProducts>
  <kineticLaw>
    <math xmlns="http://www.w3.org/1998/Math/MathML">
      <apply>
        <times/>
        <apply>
          <plus/>
          <ci>ActivateurTryptase</ci>
          <apply>
            <times/>
            <ci>moinsUn</ci>
            <ci>InhibiteurTryptase</ci>
          </apply>
        </apply>
        <ci>k3</ci>
        <ci>Tryptase</ci>
      </apply>
    </math>
    <listOfParameters>
      <parameter id="k3" value="0.27"/>
      <parameter id="un" value="1"/>
      <parameter id="moinsUn" value="-1"/>
    </listOfParameters>
  </kineticLaw>
</reaction>

<!-- ||| TNF ||| -->

<!--===== liberation du TNF-alpha préformé -->
<reaction id="FcERI_Allergene_creation" reversible="false">
  <listOfReactants>
    <speciesReference species="TnfInternePre" compartment="cytoplasme" stoichiometry="1"/>
  </listOfReactants>
  <listOfModifiers>
    <modifierSpeciesReference species="FcERI_Allergene" compartment="membrane"
      stoichiometry="1"/>
  </listOfModifiers>
  <listOfProducts>
    <speciesReference species="TNFalpha_milieu" compartment="milieu" stoichiometry="1"/>
  </listOfProducts>
  <kineticLaw>
    <math xmlns="http://www.w3.org/1998/Math/MathML">
      <apply>
        <times/>
        <ci>ka</ci>
        <ci>TnfInternePre</ci>
        <ci>FcERI_Allergene</ci>
      </apply>
    </math>
    <listOfParameters>
      <parameter id="ka" value="25000000"/>
    </listOfParameters>
  </kineticLaw>
</reaction>

```

```
<!--===== liberation du TNF-alpha néoformé -->
<reaction id="SyntheseNeoB" reversible="false">
  <listOfReactants>
    <speciesReference species="PreTnfInterneNeoA" compartment="cytoplasme" stoichiometry="1"/>
  </listOfReactants>
  <listOfModifiers>
    <modifierSpeciesReference species="FcERI_Allergene" compartment="membrane"/>
    <modifierSpeciesReference species="PreTnfInterneNeoC" compartment="cytoplasme"/>
  </listOfModifiers>
  <listOfProducts>
    <speciesReference species="PreTnfInterneNeoB" compartment="cytoplasme" stoichiometry="1"/>
  </listOfProducts>
  <kineticLaw>
    <math xmlns="http://www.w3.org/1998/Math/MathML">
      <apply>
        <times/>
        <ci>PreTnfInterneNeoA</ci>
        <apply>
          <plus/>
          <apply>
            <times/>
            <ci>kOANeo</ci>
            <ci>FcERI_Allergene</ci>
          </apply>
          <apply>
            <times/>
            <ci>PreTnfInterneNeoC</ci>
            <ci>kC</ci>
          </apply>
        </apply>
      </math>
    <listOfParameters>
      <parameter id="kC" value="800000"/>
      <parameter id="kOANeo" value="0.0000001"/>
    </listOfParameters>
  </kineticLaw>
</reaction>

<!--=====Synthese NeoC -->
<reaction id="Synthese NeoC" reversible="false">
  <listOfReactants>
    <speciesReference species="PreTnfInterneNeoB" compartment="cytoplasme" stoichiometry="1"/>
  </listOfReactants>
  <listOfProducts>
    <speciesReference species="PreTnfInterneNeoC" compartment="cytoplasme" stoichiometry="1"/>
  </listOfProducts>
  <kineticLaw>
    <math xmlns="http://www.w3.org/1998/Math/MathML">
      <apply>
        <times/>
        <ci>kOBNeo</ci>
        <ci>PreTnfInterneNeoB</ci>
      </apply>
    </math>
    <listOfParameters>
      <parameter id="kOBNeo" value="9"/>
    </listOfParameters>
  </kineticLaw>
</reaction>
```

```
<reaction id="Consommation NeoC" reversible="false">
  <listOfReactants>
    <speciesReference species="PreTnfInterneNeoC" compartment="cytoplasme" stoichiometry="1"/>
  </listOfReactants>
  <kineticLaw>
    <math xmlns="http://www.w3.org/1998/Math/MathML">
      <apply>
        <times/>
        <ci>kOCNeo</ci>
        <ci>PreTnfInterneNeoC</ci>
      </apply>
    </math>
    <listOfParameters>
      <parameter id="kOCNeo" value="0.01"/>
    </listOfParameters>
  </kineticLaw>
</reaction>

<!--=====Synthese TnfInterneNeo -->
<reaction id="Synthèse TnfInterneNeo" reversible="false">
  <listOfModifiers>
    <modifierSpeciesReference species="PreTnfInterneNeoC" compartment="cytoplasme"
                              stoichiometry="1"/>
  </listOfModifiers>
  <listOfProducts>
    <speciesReference species="TnfInterneNeo" compartment="cytoplasme" stoichiometry="1"/>
  </listOfProducts>
  <kineticLaw>
    <math xmlns="http://www.w3.org/1998/Math/MathML">
      <apply>
        <times/>
        <ci>k1Neo</ci>
        <ci>PreTnfInterneNeoC</ci>
      </apply>
    </math>
    <listOfParameters>
      <parameter id="k1Neo" value="0.008"/>
    </listOfParameters>
  </kineticLaw>
</reaction>
```

```
<!--=====TNFNeoformé -->
<!-- liberation du TNF-alpha neoformé -->
<reaction id="LiberationPre" reversible="false">
  <listOfReactants>
    <speciesReference species="TnfInterneNeo" compartment="cytoplasme"
                      stoichiometry="1"/>
  </listOfReactants>
  <listOfModifiers>
    <modifierSpeciesReference species="FcERI_Allergene" compartment="plasma"/>
  </listOfModifiers>
  <listOfProducts>
    <speciesReference species="TNFalpha_milieu" compartment="plasma" stoichiometry="1"/>
  </listOfProducts>
  <kineticLaw>
    <math xmlns="http://www.w3.org/1998/Math/MathML">
      <apply>
        <times/>
        <ci>k3neo</ci>
        <ci>TnfInterneNeo</ci>
        <ci>FcERI_Allergene</ci>
      </apply>
    </math>
    <listOfParameters>
      <parameter id="k3neo" value="4200000"/>
    </listOfParameters>
  </kineticLaw>
</reaction>

  <!-- recepteur internalisation ... -->
</listOfReactions>
</model>
</sbml>
```

Annexe B

Diagrammes de classes UML

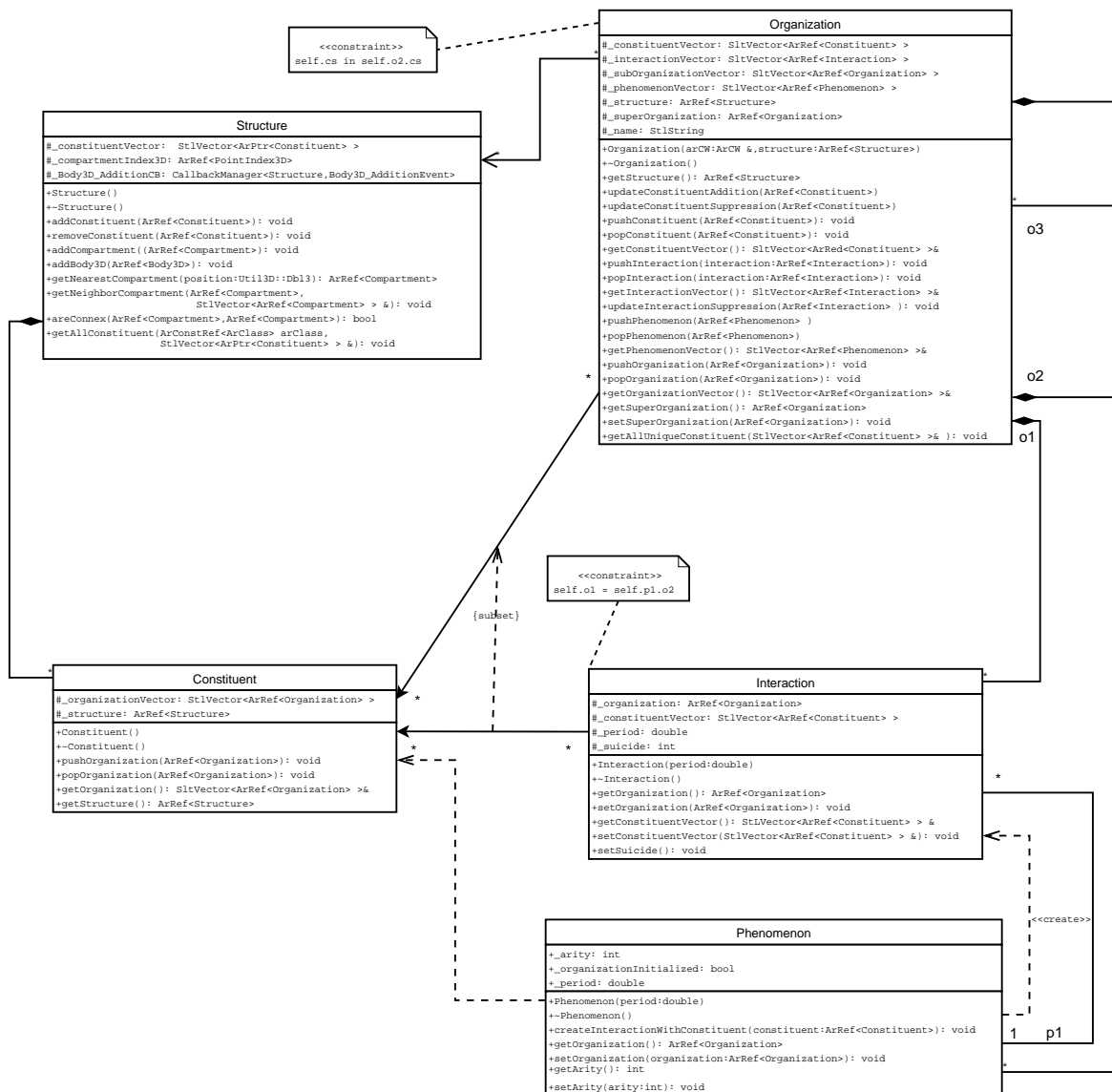


FIG. B.1: Détail des classes de base de la libBio.

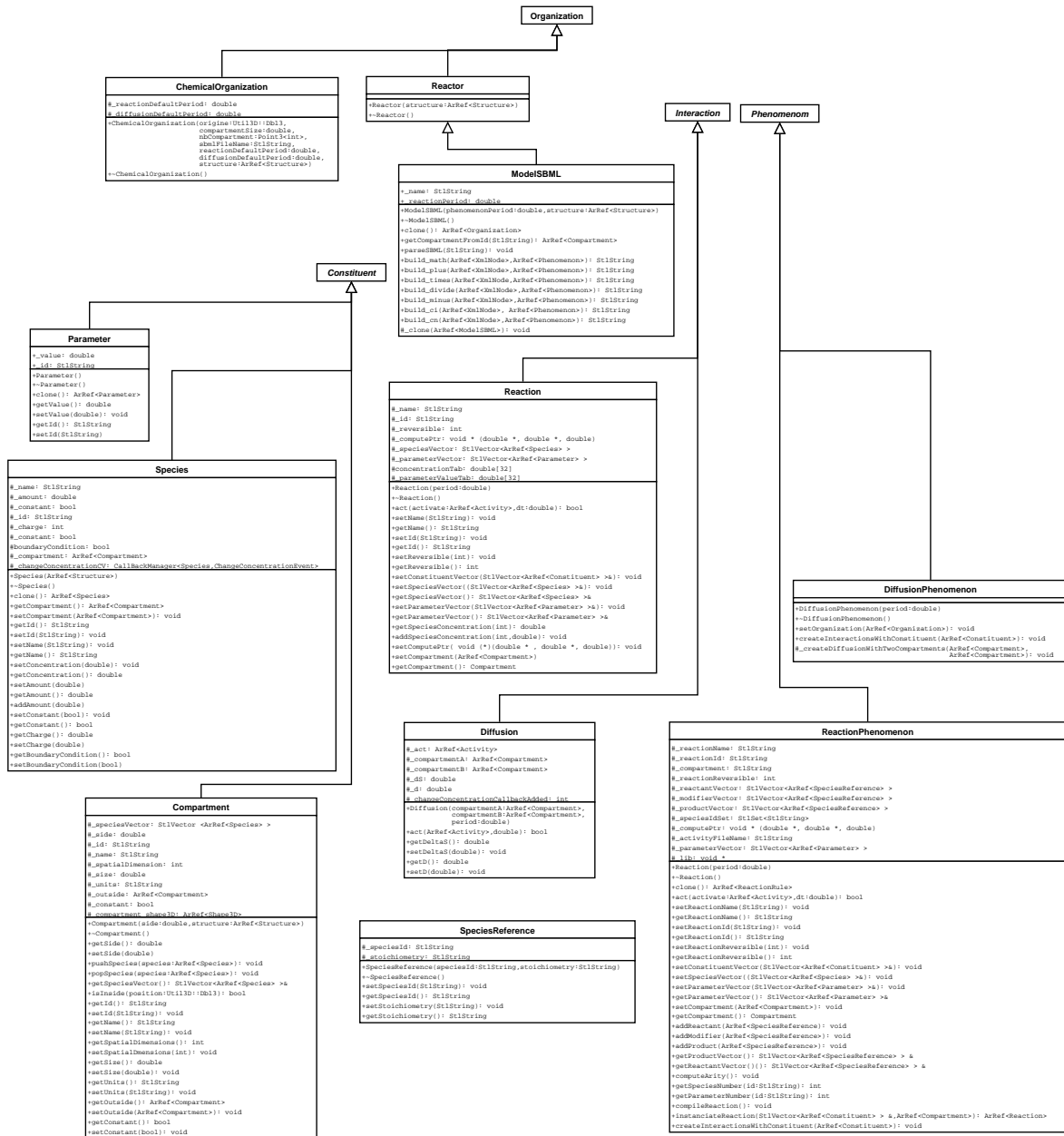


FIG. B.2: Classes liées au phénomènes de nature chimique. Les dépendances entre les classes sont omises.

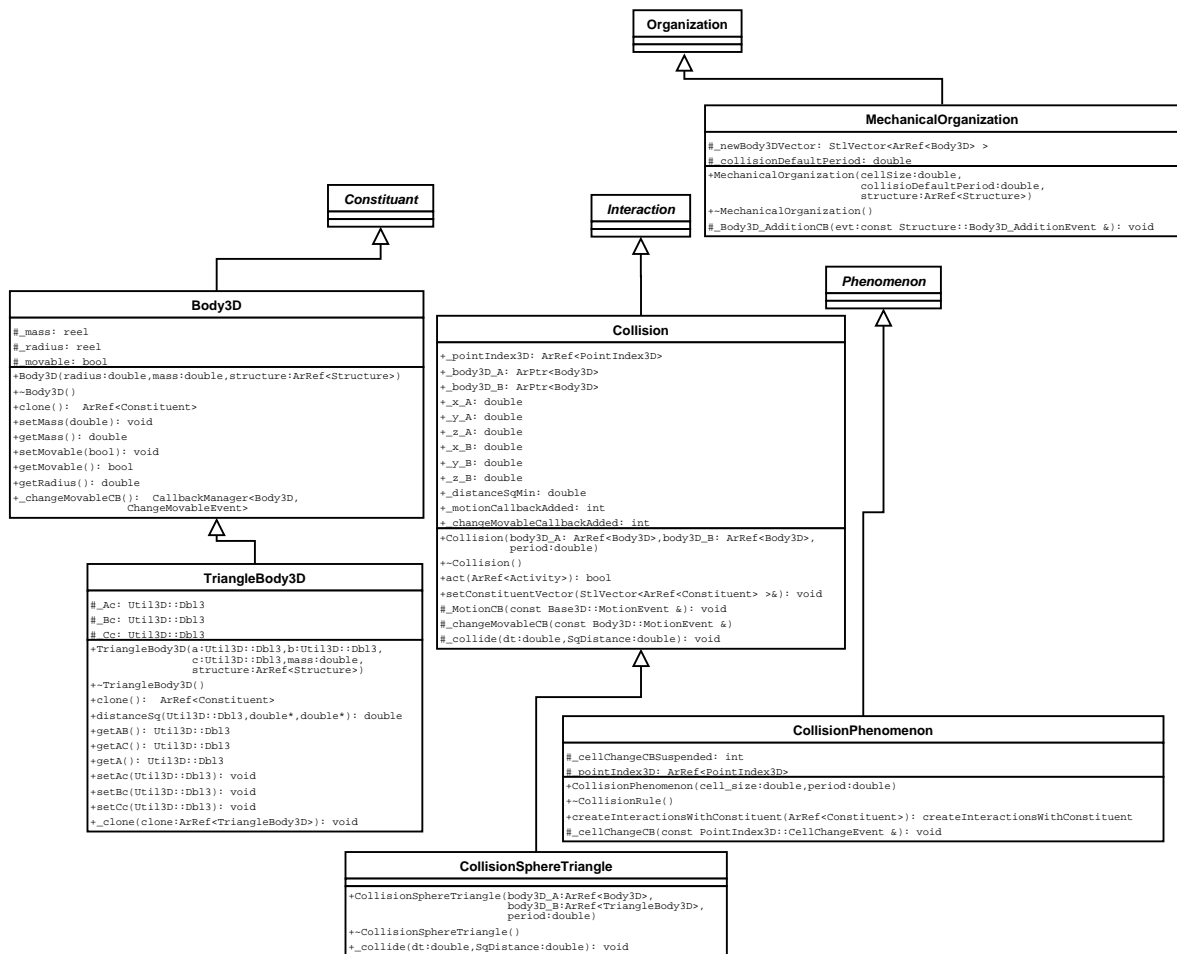


FIG. B.3: Classe liées au phénomènes de collisions. Les dépendances entre les classes sont omises.

Annexe C

Résumé long

Ce travail s'inscrit dans l'histoire du développement d'un nouveau mode d'investigation du vivant : l'expérimentation *in virtuo*.

Nous l'introduisons en présentant successivement trois niveaux de prise en compte de la complexité biologique : le changement de paradigme en biologie, la modélisation de cette complexité en informatique et finalement, le recours au domaine de la réalité virtuelle pour une production de connaissances basée sur l'expérience des modèles numériques.

Nous proposons un cadre générique de modélisation et de simulation multi-modèles et multi-échelles, permettant l'expérience *in virtuo*. Nous en décrivons alors une implémentation fournissant une boîte à outils pour la modélisation des systèmes biologiques.

Enfin, les applications réalisées dans le cadre plus large d'une collaboration avec une équipe de biologistes seront exposées.

1- En premier lieu, nous constatons que l'émergence de la biologie moléculaire a provoqué une révolution en biologie. D'une part, la découverte de la molécule d'ADN, et de l'autre les apports humains, techniques, épistémologiques et théoriques des sciences de la physique et de la chimie ont conduit à une explosion des données biologiques. La biologie, science expérimentale, manque cruellement d'outils théoriques nécessaires à leurs analyses et à celles des systèmes complexes sous-jacents qu'elles décrivent. Ainsi, à l'instar de ce qui s'est passé en sciences physiques au début du siècle dernier, un changement de paradigme s'opère aujourd'hui en biologie pour donner naissance aux biologies « théorique », « systémique » ou « intégrative ». Ces nouvelles approches théoriques sont indissociables de la modélisation et font appel aux concepts de système, d'interaction, de rétroaction, de régulation, d'organisation, d'évolution, etc...

2 - L'informatique en temps qu'outils de modélisation et de simulation, est indispensable à l'étude du vivant. L'appellation *in silico* désigne alors les travaux menés dans ce domaine. Il est possible de distinguer deux grands types d'étude *in silico* des systèmes vivants.

D'abord, les mathématiques et les méthodes formelles de l'informatique permettent l'analyse *a priori* des modèles. Parmi ces formalismes, les algèbres de processus, les réseaux de Petri et la logique temporelle permettent de manipuler les réseaux complexes de la biologie.

Ensuite, la simulation permet la prédiction et l'explication *a posteriori*. Dans cette dernière approche à laquelle nous nous intéressons plus particulièrement, plusieurs méthodes se distinguent. Classiquement sont utilisées les méthodes conventionnelles basées sur des algorithmes de résolution d'équations différentielles ordinaires ou aux dérivées partielles. Plus

récemment, les méthodes de programmation orientées objets et les simulations individu-centrées ont été mises à contribution. Dans ce cadre, les systèmes multi-agents se montrent extrêmement utiles. L'intérêt de ces outils est qu'ils tendent à offrir des possibilités de modélisation et de simulation : modulaires, multi-échelles et multi-modèles.

3 - L'expression « expérimentation *in silico* » est parfois utilisée dans les articles relatant de biologie *in silico*. Nous nous interrogeons sur cette notion d'expérimentation, et plus globalement sur celle d'expérience. Pour se faire, il apparaît intéressant de nous référer à la psychologie cognitive qui, elle aussi, est sujette à un changement de paradigme du cognitivisme, vers le constructivisme. En effet, les récentes avancées de ce domaine mettent en exergue l'*énaction* comme théorie permettant de dépasser la vision désuète du cognitivisme (sur laquelle se sont construits un bon nombre des outils de l'intelligence artificielle). Ainsi nous précisons le lien entre expérience et connaissance.

Les concepts de la réalité virtuelle et le lien fort que ce domaine entretient avec l'acte d'expérience nous permettent alors d'envisager un nouveau type d'expérimentation des modèles numériques : l'expérience *in virtuo*. Cette dernière passe nécessairement par la double médiation des sens et de l'action. Nous montrons ensuite que le principe d'autonomie se révèle incontournable pour construire les univers virtuels. Se faisant, la réalité virtuelle offre également un troisième type de médiation : la médiation de l'esprit. L'utilisateur/modélisateur peut alors faire l'expérience d'univers virtuels, tester de nouvelles hypothèses ou simplement mieux l'appréhender à travers l'expérience *in virtuo*.

Les différents travaux réalisés au Centre Européen de Réalité Virtuelle (CERV) tendent à mettre en pratique les concepts (immersion, interaction, autonomie) et les outils de la réalité virtuelle (ARéVi) pour modéliser et expérimenter les systèmes complexes vivants. Ainsi nous présentons les modèles d'agents **Cellule** (Pascal Ballet), **Réaction** (Sébastien Kerdélo) et l'introduction de la notion d'organisation (Gabriel Querrec) qui y ont été développés.

4 - La ré-appropriation des travaux du CERV concernant la biologie, nous amène à exprimer le besoin d'un cadre générique de modélisation qui soit cohérent, modulaire, réutilisable, multi-échelles et multi-modèles pour l'étude des systèmes vivants. Ce travail de généralisation propose alors une vision de la modélisation centrée sur les interactions plutôt que sur les composants. Pour reprendre l'exemple de la chimie, cela revient à modéliser les réactions chimiques (agents **Réaction**) plutôt que les molécules. Nous sommes alors amenés à réifier le concept d'interaction en entités actives autonomes peuplant nos univers virtuels. Cette démarche propose donc un changement de repère de modélisation. A l'instar du traitement du signal qui utilise un repère fréquentiel plutôt que temporel pour étudier les signaux périodiques, nous utilisons une modélisation basée interaction plutôt que composant pour modéliser et faire vivre les systèmes complexes du vivant.

Nous complétons ensuite le modèle en proposant :

- un ordonnancement multi-échelles, chaotique, asynchrone (avec et sans remise) pour nos interactions autonomes. Cet ordonnancement repose sur le principe d'un ordonnancement à événements discrets de modèles réalisant des intervalles de temps ;
- la réification du concept de phénomène en un objet permettant la création des objets actifs **Interaction** comme réalisations particulières de ce même phénomène ;
- l'organisation des modèles en systèmes autonomes opérationnellement clos et en couplage structurel les uns avec les autres. La topologie structurelle de chaque système est alors maintenue selon les règles subjectives données par le modélisateur et par les différentes organisations associées au système. Cette aspect du modèle est fortement inspiré des travaux de Varela (également auteur de l'*énaction*) sur la modélisation du

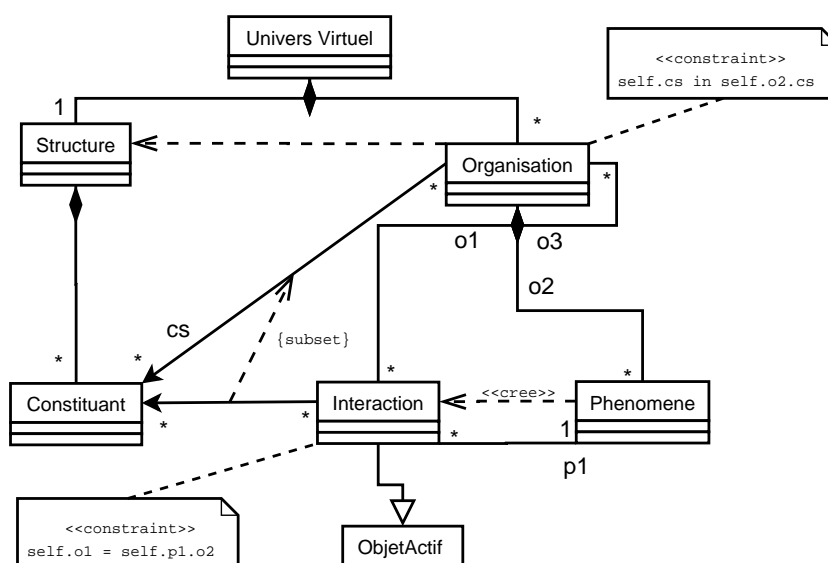


FIG. C.1: Diagramme de classe du modèle générique.

vivant.

Finalement, nous obtenons un modèle générique basé sur 5 classes de base : **Interaction** (objet actif), **Phénomène**, **Organisation**, **Structure** et **Constituant**, à partir duquel nous pouvons modéliser une grande variété de phénomènes.

5 - Le modèle générique étant défini, nous montrons alors comment il peut être dérivé en modèles de natures très différentes. Nous les avons rassemblés au sein d'une « boîte à outils » pour la modélisation de systèmes biologiques.

Tout d'abord, nous reprenons les notions d'agents **Réaction** et **Diffusion** pour la définition d'organisations chimiques. L'utilisation de la norme SBML permet la définition et la récupération de toutes sortes de réseaux biochimiques.

Ainsi, nous pouvons composer avec les modèles chimiques pour construire un modèle chimique de cellule. L'organisation de la cellule définit alors un système autonome en couplage avec son milieu.

Ensuite nous dotons l'organisation de la cellule d'une composante mécanique : un corps en trois dimensions. L'ensemble des corps peuvent alors interagir les uns avec les autres au sein d'une autre organisation composée d'un phénomène réalisant les collisions mécaniques. Nous montrons alors comment la gestion des collisions peut être traitée comme n'importe quel autre phénomène par le modèle générique.

Enfin nous discutons de la place du modèle de l'utilisateur pour l'expérience *in virtuo*. La prise en compte de l'utilisateur perçu comme un système autonome parmi les autres est alors immédiate.

6 - Cette thèse s'inscrit dans le cadre d'une collaboration avec une équipe de recherche en neurobiologie cutanée, dont l'objet d'étude est l'interaction entre la peau, le système immunitaire et le système nerveux. Nous présentons un modèle de peau (un cube de $1m^3$) permettant l'expérimentation *in virtuo* du phénomène d'urticaire allergique.

La réalisation de cette application a permis la mise en œuvre, dans un cas d'utilisation concret, du modèle générique ainsi que de la « boîte à outils » réalisée pendant cette thèse. Se faisant, elle montre que le cadre de modélisation proposé possède une grande expressivité

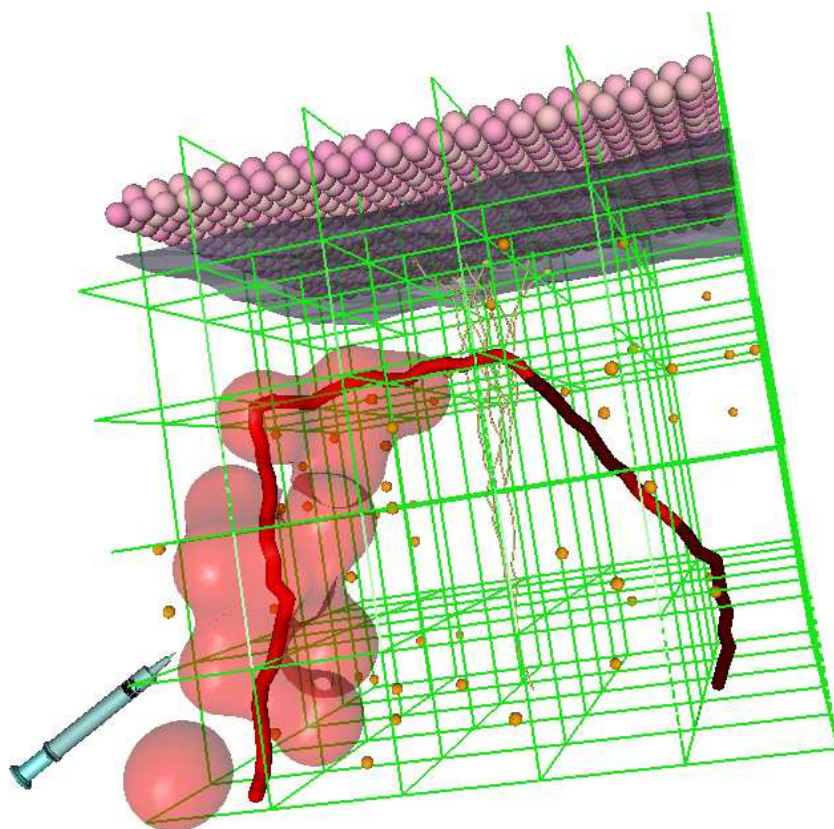


FIG. C.2: Capture d'écran de l'application.

et permet la réalisation, en temps réel, de simulations assez conséquentes pour présenter un intérêt. L'aspect modulaire et réutilisable du code généré est un atout de plus de la démarche.

D'un point de vue biologique, l'application permet de rassembler et de manipuler l'état des connaissances à travers l'exploitation du modèle numérique. Nous constatons également les possibilités interdisciplinaires et pédagogiques de telles applications.

7 - Pour terminer cet exposé, nous présentons deux applications supplémentaires (en cours de réalisation) dont les principes permettent de montrer d'autres possibilités du modèle générique.

Premièrement, nous décrivons un projet de maillage de l'espace basé sur un diagramme de Voronoï pour la diffusion moléculaire. La méthode permet aux modèles numériques peuplant l'univers virtuel de structurer eux-mêmes leur environnement, ce qui présente un pas supplémentaire vers l'autonomisation des modèles et vers l'allègement des calculs.

Deuxièmement, nous exposons un projet de modélisation de systèmes autopoïétiques imaginaires inspirés du fonctionnement multi cellulaire. En posant de nouvelles questions, cette démarche permet d'élargir le champ d'utilisation du modèle générique : de la modélisation de systèmes du vivant vers la modélisation de systèmes vivants.

Ainsi, cette thèse, représente une étape dans la construction d'un véritable laboratoire virtuel pour l'expérimentation *in virtuo*. Nous avons choisi une posture de modélisation originale qui met l'accent sur les phénomènes et les interactions. Nous pouvons la considérer comme une variante aux systèmes multi-agents réactifs. Cette méthode est adaptée à l'étude des systèmes complexes biologiques, et est intrinsèquement construite pour permettre l'interaction avec l'utilisateur/modélisateur pendant l'exécution. Le travail concernant les interfaces constituera une des étapes suivantes de ce travail qui révélera plus largement l'intérêt méthodologique de l'interaction avec l'utilisateur. Toutefois, les applications réalisées montrent déjà les qualités ergonomiques et la grande expressivité de la méthode.

Bibliographie

- Akimoto, S., Ishikawa, O., Igarashi, Y., Kurosawa, M., and Miyachi, Y. (1998). Dermal mast cells in scleroderma : their skin density, tryptase/chymase phenotypes and degranulation. *Br. J. Dermatol.*, 138(3) :399–406.
- Altschul, S., Gish, W., Miller, W., Myers, E., and Lipman, D. (1990). Basic local alignment search tool. *J Mol Biol*, 215(3) :406–10.
- Alur, R., Belta, C., Ivancic, F., Kumar, V., Rubin, H., Schug, J., Sokolsky, O., and Webb, J. (2002). Visual programming for modeling and simulation of biomolecular regulatory networks. In *HiPC '02 : Proceedings of the 9th International Conference on High Performance Computing*, pages 702–712, London, UK. Springer-Verlag.
- Amar, P., Bernot, G., and Norris, V. (2004). Hsim : a simulation programme to study large assemblies of proteins. *J Biol Physics Chem*, 4(79-84).
- Ander, M., Beltrao, P., Di Ventura, B., Ferkinghoff-Borg, J., Foglierini, M., Kaplan, A., Lemerle, C., Tomás-Oliveira, I., and Serrano, L. (2004). SmartCell, a framework to simulate cellular processes that combines stochastic approximation with diffusion and localisation : analysis of simple networks. *Systems biology*, 1(1) :129–138.
- Arkin, A., Ross, J., and McAdams, H. H. (1998). Stochastic kinetic analysis of developmental pathway bifurcation in phage lambda-infected Escherichia coli cells. *Genetics*, 149(4) :1633–48.
- Arnold, V. (1981). *Équations différentielles ordinaires*. Mir Moscou.
- Ascher, U. M. and Petzold, L. R. (1998). *Computer methods for ordinary differential equations and differential-algebraic equations*. Society for industrial and applied mathematics.
- Astbury, W. (1961). Molecular biology or ultrastructural biology ? *Nature*, 190 :1124.
- Auffray, C., Chen, Z., Hood, L., Soares, B., and Sugano, S. (2003a). Foreword : from the TRANSCRIPTOME conferences to the SYSTEMOSCOPE international consortium. *C R Biol*, 326(10-11) :867–75.
- Auffray, C., Imbeaud, S., Roux-Rouquie, M., and Hood, L. (2003b). From functional genomics to systems biology : concepts and practices. *C R Biol*, 326(10-11) :879–92.
- Bairoch, A., Apweiler, R., Wu, C., Barker, W., Boeckmann, B., Ferro, S., Gasteiger, E., Huang, H., Lopez, R., Magrane, M., Martin, M., Natale, D., O'Donovan, C., Redaschi, N., and Yeh, L. (2006). The universal protein resource (uniprot). *Nucleic Acids Res*, 34(Database issue) :D187–D191.
- Ballet, P. (2000). *Intérêts Mutuels Des Systèmes Multi-Agents Et De l'Immunologie*. PhD thesis, Université De Bretagne Occidentale, Brest.

- Ballet, P., Zemirline, A., and Marcé, L. (2004). The biodyn language and simulator. application to an immune response and e.coli and phage interaction. *Journal of Biological Physics and Chemistry*, 4 :93–100.
- Beadle, G. and Tatum, E. (1941). Genetic control of biochemical reactions. *Neurospora. Proc. Natl. Acad. Sci USA*, 27 :499–506.
- Bellman, R. (1957). *Dynamic Programming*. Princeton University Press.
- Benkirane, R. (2002). *La complexité, vertiges et promesses*, chapter Autopoïese et émergence, entretien avec Francisco Varela. Le Pommier.
- Benson, D., Karsch-Mizrachi, I., Lipman, D., Ostell, J., and Wheeler, D. (2006). Genbank. *Nucleic Acids Res*, 34(Database issue) :D16–20.
- Bernard, C. (1865). *Introduction à l'étude de la médecine expérimentale*. J. B. Baillière et fils.
- Bock, R. and Goode, J., editors (1998). *The limits of reductionism in biology*, pages 106–116. Eds. (Novartis Found. Sum., London, john wiley edition.
- Bonabeau, E., Theraulaz, G., Deneubourg, J., Aron, S., and Camazine, S. (1997). Self-organisation in social insects. *Trends in Ecology and Evolution (Elsevier)*, 12(5) :188–193.
- Bosse, T., Jonker, C. M., and Treur, J. (2006). Simulation and analysis of complex biological processes : an organisation modelling perspective. In *Annual Simulation Symposium*, pages 304–311.
- Bower, J. M. and Beeman, D. (1998). *The book of Genesis : exploring realistic neural models with the general neural simulation system*. Springer.
- Braverman, I. (2000). The cutaneous microcirculation. *JID Symposium proceedings*, 5(1) :3 – 9.
- Broderick, G., Rúaini, M., Chan, E., and Ellison, M. J. (2004). A life-like virtual cell membrane using discrete automata. In *Silico Biology*, 5.
- Buche, C., Querrec, R., De Loor, P., and Chevaillier, P. (2004). MASCARET : A pedagogical multi-agent system for virtual environment for training. *Special issue "Cyberworlds and education" of the international Journal of Distance Education Technologies (JDET)*, 2(4) :41–61.
- Cadoz, C. (1994). *Les réalités virtuelles*. Flammarion, Paris, collection dominos edition.
- Camp, J. J., Cameron, B. M., and Blezek, D. and Robb, R. A. (1998). Virtual reality in medicine and biology. *Future Generation Computer Systems*, 14(1-2) :91–108.
- Cao, Y., Li, H., and Petzold, L. R. (2004). Efficient formulation of the stochastic simulation algorithm for chemically reacting systems. *The journal of chemical physics*, 121(9) :4059–4067.
- Capra, F. (1975). *Le Tao de la physique*. Sand.
- Capra, F. (2003). *La toile de la vie*. Edition du Rocher.
- Cardelli, L. (2004). Brane calculi. In *CMSB*, pages 257–278.
- Cardelli, L. G. A. (2000). Mobile ambients. *Theoretical Computer Science*, 240(1) :177–213.
- Chen, M., Freier, A., Köhler, J., and Ruegg, A. (2002). The biology petri net markup language. In *Promise*, pages 150–161.
- Church, M. and Clough, G. (1999). Human skin mast cells : in vitro and in vivo studies. *Ann Allergy Asthma Immunol*, 83(5) :471.
- Danos, V. and Laneve, C. (2004). Formal molecular biology. *Theor. Comput. Sci.*, 325(1) :69–110.
- De Rosnay, J. (1975). *Le microscope : Vers une vision globale*. Le Seuil.

- Demazeau, Y. (1995). From interactions to collective behaviour in agent-based systems. In *First European conference on cognitive science*, pages 117–132, Saint Malo, France.
- Desmeulles, G. (2003). Simulation cellulaire *in virtuo*, perspectives multi agents à travers la réification des interactions. Master’s thesis, Rapport de stage de D.E.A.. Institut de Formation Supérieurs en Informatique et Communication (IFSIC), Université de Rennes I, Rennes.
- Desmeulles, G., Dupré, D., Rodin, V., and Misery, L. (2003a). La simulation informatique multi-agents : Vers une nouvelle voie de recherche en allergologie et en immunodermatologie. In *CARD’2003, Congrès Annuel de la Recherche Dermatologique*, page 83, Rouen (France). Poster P31.
- Desmeulles, G., Dupré, D., Rodin, V., and Misery, L. (2003b). Mise au point d’un modèle de recherche *in virtuo* de la réaction allergique. In *JDP’2003, Journées Dermatologiques de Paris. Supplément des annales de dermatologie et de vénéréologie, Masson Editeur*, volume 130, pages 71–72, Paris (France). Communication orale C110.
- Desmeulles, G., Dupré, D., Rodin, V., and Misery, L. (2003c). Multi-agents computer simulation : towards a new way of research in allergy and immunodermatology. *Journal of Investigative Dermatology*, 121(6) :1570.
- Desmeulles, G., Dupré, D., Rodin, V., and Misery, L. (2005a). Mise au point d’un modèle de recherche *in virtuo* sur la réaction allergique. *Annales de Dermatologie et de Vénéréologie*, 132(8–9) :697–701.
- Desmeulles, G., Misery, L., and Rodin, V. (2005b). Modeling systems biology for *in virtuo* experiments. In *Ecole thématique : Modélisation de systèmes biologiques complexes dans le contexte de la génomique*, pages 119–120, Montpellier.
- Desmeulles, G., Querrec, G., Redou, P., Kerdélo, S., Misery, L., Rodin, V., and Tisseau, J. (2006). The virtual reality applied to the biology understanding : the *in virtuo* experimentation. *Expert Systems with Applications*, 30(1) :82–92.
- Desmeulles, G., Redou, P., Misery, L., and Rodin, V. (2005c). A multiagent and multimodel approach for the experimentation on biological complex systems. In *WMSCI’2005, 9th World Multi-Conference on Systemics, Cybernetics and Informatics*, volume 6, pages 142–147, Orlando, Florida (USA).
- Desmeulles, G., Rodin, V., and Misery, L. (2005d). Modèle *in virtuo* d’urticaire allergique : résultats préliminaires. In *CARD’2005, Congrès Annuel de la Recherche Dermatologique*, page 18, Brest (France). Communication orale C12.
- Dhar, P., Meng, T. C., Somani, S., Ye, L., Sairam, A., Chitre, M., Hao, Z., and Sakharkar, K. (2004). Cellware – a multi-algorithmic software for computational systems biology. *Bioinformatics*, 20(8) :1319–1321.
- Doi, A., Fujita, S., Matsuno, H., Nagasaki, M., and Miyano, S. (2004). Constructing biological pathway models with hybrid functional Petri nets. *In Silico Biologie*, 4(3) :271–91.
- Dorin, A. (2002). The visual aspect of artificial life. In Tanaka, S. ., editor, *Proceedings of the 7th International Symposium on Artificial Life and Robotics*, pages 451–455, Beppu, Japan.
- Ehlde, M. and Zacchi, G. (1995). Mist : a user-friendly metabolic simulator. *Computer applications in the biosciences*, 11(2) :201–207.
- Elf, J. and Ehrenberg, M. (2004). Spontaneous separation of bi-stable biochemical systems into spatial domains of opposite phases. *Systems biology*, 1(2) :230–236.
- Ellis, S. (1991). Nature and origin of virtual environments : a bibliographic essay. *Computing Systems in Engineering*, 2(4) :321–347.
- Erman, L. D., Hayes-Roth, F., Lesser, V. R., and R., R. D. (1980). The hearsay-ii speech-understanding system : Integrating knowledge to resolve uncertainty. *ACM Comput. Surv.*, 12(2) :213–253.
- Ermentrout, B. (2002). *Simulating, analyzing, and animating dynamical systems : a guide to Xppaut for researchers and students*. Society for industrial and applied mathematics.

- Fages, F., Soliman, S., and Chabrier-Rivier, N. (2004). Modelling and querying interaction networks in the biochemical abstract machine biocham. *Journal of Biological Physics and Chemistry*, 4 :64–73.
- Ferber, J. (1997). *Les Systèmes Multi Agents*. InterEdition.
- Ferber, J. and Gutknecht, O. (1998). A meta-model for the analysis and design of organizations in multi-agent systems. In *ICMAS '98 : Proceedings of the 3rd International Conference on Multi Agent Systems*, page 128, Washington, DC, USA. IEEE Computer Society.
- Ferber, J., Gutknecht, O., and Michel, F. (2004). From agents to organizations : an organizational view of multi-agent systems. In *Agent-Oriented Software Engineering IV 4th International Workshop*, pages 214–230, Melbourne, Australia. Springer Verlag.
- Finney, A. and Hucka, M. (2003). Systems biology markup language : Level 2 and beyond. *Biochem. Soc. Trans.*, 31(Pt 6) :1472–3.
- Froment, N. (2005). Développement de l'application derme in virtuo. Master's thesis, Master CCI, IFSIC.
- Fuchs, P., Arnaldi, B., Moreau, G., Burkhardt, J.-M., Chauffaut, A., Coquillart, S., Donikian, S., Duval, T., Grosjean, J., Harrouet, F., Klinger, E., Lourdeaux, D., Mellet d'Huart, D., Paljic, A., Papin, J.-P., Stergiopoulos, P., Tisseau, J., and Viaud-Delmon, I., editors (2001). *Le traité de la réalité virtuelle*, volume 1, chapter 1, pages 1–35. Les Presses de l'Ecole des Mines.
- Fuchs, P., Moreau, G., Arnaldi, B., Burkhardt, J., Chauffaut, A. and Coquillart, S., Donikian, S., Duval, T., Grosjean, J., Harrouet, F., Klinger, E., Lourdeaux, D., Mellet d'Huart, D., Paljic, A., Papin, J. P., Stergiopoulos, P., Tisseau, J., and Viaud-Delmon, I., editors (2003). *Le traité de la réalité virtuelle*, volume 1. Les Presses de l'École des Mines de Paris, deuxième édition. ISBN : 2-911762-47-9.
- Fuchs, P., Moreau, G., and Tisseau, J. (2006). *Le traité de la réalité virtuelle*, volume 3. Presses de l'école des Mines.
- Galperin, M. (2006). The molecular biology database collection : 2006 update. *Nucleic Acids Res*, 34(Database issue).
- Gibson, M. A. and Bruck, J. (2000). Efficient exact stochastic simulation of chemical systems with many species and many channels. *The journal of physical chemistry A*, 104(9) :1876–1889.
- Gibson, W. (1980). *Neuromancer*. ACE Books, New York.
- Gillespie, D. (1976). A general method for numerically simulating the stochastic time evolution of coupled chemical reaction. *Journal of Computational Physics*, 22 :403–437.
- González, P., Cárdenas, M., Camacho, D., Franyuti, A., Rosas, O., and Lagúnez-Otero, J. (2003). Cellulat : an agent-based intracellular signalling model. *BioSystems*, 68 :171–185.
- Goryanin, I., Hodgman, T. C., and Selkov, E. (1999). Mathematical simulation and analysis of cellular metabolism and regulation. *Bioinformatics*, 15(9) :749–758.
- Hairer, E., Norsett, S., and Wanner, G. (1993). *Solving ordinary differential equations I. Nonstiff problems*, volume 8 of *Springer series in computational mathematics*. Springer, 2ème édition.
- Hairer, E. and Wanner, G. (1996). *Solving ordinary differential equations II. Stiff and differential-algebraic problems*, volume 14 of *Springer series in computational mathematics*. Springer, 2ème édition.
- Harel, D. (1987). Statecharts : a visual formalism for complex systems. *Science of Computer Programming*, 8(3) :231–274.
- Harrouet, F. (6 juin 2001). *oRis : s'Immerger Par Le Langage Pour Le Prototypage d'Univers Virtuels À Base d'Entités Autonomes*. Thèse, Université de Bretagne Occidentale, Brest, <http://www.enib.fr/~harrouet>.
- Harrouet, F., Cazeaux, E., and Jourdan, T. (2006). ARéVi. In Fuchs, P., Moreau, G., and Tisseau, J., editors, *Le traité de la Réalité Virtuelle*, volume 3, pages 369–392. Les Presses de l'Ecole des Mines, third edition.

-
- Harrouet, F., Tisseau, J., P., R., and Chevaillier, P. (2002). oRis : un environnement de simulation interactive multi-agents. *Revue des Sciences et Technologies de l'Information, série Technique et Science Informatiques (RSTI-TSI)*, (4 - Vol 21) :499-524.
- Hewitt, C. (1977). Viewing control structures as patterns of passing messages. *Artificial Intelligence*, 8(3) :323-364.
- Hines, M. L. and Carnevale, N. T. (1997). The NEURON simulation environment. *Neural computation*, 9(6) :1179-1209.
- Holloway, R., Fuchs, H., and Robinett, W. (1992). Virtual-worlds research at the university of north carolina at chapel hill. In *Computer Graphics'92*, volume 15, pages 1-10.
- Holzhütter, H. G. and Colosimo, A. (1990). SIMFIT : a microcomputer software-toolkit for modelistic studies in biochemistry. *Computer applications in the biosciences*, 6(1) :23-28.
- Hucka, M. and Finney, A. B. B. J., Keating, S. M., Shapiro, B. E., Matthews, J. and Kovitz, B., Schilstra, M., Funahashi, A., Doyle, J. C., and Kitano, H. (2004). Evolving a lingua franca and associated software infrastructure for computational systems biology : the systems biology markup language (sbml) project. *Systems Biology*, 1(1) :41-53.
- Hucka, M., Finney, A., Sauro, H., Bolouri, H., Doyle, J., Kitano, H., the rest of the SBML Forum : Arkin, A., Bornstein, B. J., Bray, D., Cornish-Bowden, A., Cuellar, A. A., Dronov, S., Gilles, E. D., Ginkel, M., Gor, V., Goryanin, I. I., Hedley, W. J., Hodgman, T. C., Hofmeyr, J.-H., Hunter, P. J., Juty, N. S., Kasberger, J. L., Kremling, A., Kummer, U., Le Novère, N., Loew, L. M., Lucio, D., Mendes, P., Minch, E., Mjolsness, E. D., Nakayama, Y., Nelson, M. R., Nielsen, P. F., Sakurada, T., Schaff, J. C., Shapiro, B. E., Shimizu, T. S., Spence, H. D., Stelling, J., Takahashi, K., Tomita, M., Wagner, J., and Wang, J. (2003). The systems biology markup language (sbml) : a medium for representation and exchange of biochemical network models. *Bioinformatics*, 19(4) :524-531.
- Hunter, P. J. and Borg, T. K. (2003). Integration from protein to organs : The physiome project. *Nature Review Molecular Cell Biology*.
- Hübner, J. and Sichman, J.S. and Boissier, O. (2002). Spécification structurelle, fonctionnelle et déontique d'organisations dans les systèmes multi-agents. In *JFIAD-SMA'02*.
- Ichikawa, K. (2001). A-Cell : graphical user interface for the construction of biochemical reaction models. *Bioinformatics*, 17(5) :483-484.
- Javier Vázquez-Salceda, Virginia Dignum, F. D. (2004). Organizing multi-agent systems. Technical Report UU-CS-2004-015, Institute of Information and Computing Sciences, Utrecht University.
- Johnson, C. G., Goldmen, P. J., and Gullick, W. J. (2003). Simulating complex intracellular processes using object-oriented computational modelling. *Progress in Biophysics & Molecular Biology*.
- Kam, N., Cohen, I., and Harel, D. (2001). The immune system as a reactive system : Modeling t-cell activation with statecharts. In *Proceedings of Symposium on Visual Languages and Formal Methods, IEEE* , pages 15-22.
- Kanehisa, M., Goto, S., Hattori, M., Aoki-Kinoshita, K. F., Itoh, M., Kawashima, S., Katayama, T., Araki, M., and Hirakawa, M. (2006). From genomics to chemical genomics : new developments in KEGG. *Nucleic Acids Res*, 34(Database issue) :D354-7.
- Kanz, C., Aldebert, P., Althorpe, N., Baker, W., Baldwin, A., Bates, K., Browne, P., van den Broek, A., Castro, M., Cochrane, G., Duggan, K., Eberhardt, R., Faruque, N., Gamble, J., Diez, F., Harte, N., Kulikova, T., Lin, Q., Lombard, V., Lopez, R., Mancuso, R., McHale, M., Nardone, F., Silventoinen, V., Sobhany, S., Stoehr, P., Tuli, M., Tzouvara, K., Vaughan, R., Wu, D., Zhu, W., and Apweiler, R. (2005). The embl nucleotide sequence database. *Nucleic Acids Res.*, 33(Database issue) :D29-33.
- Karr, T. L. and Brady, R. (2000). Virtual biology in the CAVE. *Trends Genet*, 16(5) :231-2.
- Kastin, S. and Wexler, J. (1998). Bioinformatics : searching the Net. *Semin Nucl Med*, 28(2) :177-87.

- Kerdélo, S. (2006). *Méthodes informatiques pour l'expérimentation in virtuo de la cinétique biochimique. Application à la coagulation du sang*. PhD thesis, Université de Bretagne Occidentale.
- Kitano, H. (2002). Computational systems biology. *Nature*, 420(6912) :206–10.
- Kitano, H. (2003). A graphical notation for biochemical networks. *Biosilico*.
- Kitano, H., Funahashi, A., Matsuoka, Y., and Oda, K. (2005). Using process diagrams for the graphical representation of biological networks. *Nature Biotechnology*, 23 :2005.
- Krueger, M. (1983). *Artificial Reality*. Addison-Wesley, New York.
- Kuhn, T. (1972). *La Structure des Révolutions Scientifiques*. Flammarion.
- Kumar, S. P. and Feidler, J. (2003). BioSPICE : a computational infrastructure for integrative biology. *OMICS*, 7(3) :225.
- Kupiec, J. and Sonigo, P. (2000). *Ni Dieu ni gène : Pour une autre théorie de l'hérédité*. Seuil.
- Kuttler, C. and Niehren, J. (2006). Gene regulation in the pi calculus : simulating cooperativity at the lambda switch. *Transactions on Computational Systems Biology*.
- Kuzmic, P. (1996). Program dynafit for the analysis of enzyme kinetic data : application to HIV proteinase. *Analytical biochemistry*, 237(2) :260–273.
- Lales, C., Parisey, N., Mazat, J.-P., and Beurton-Aimar, M. (2005). Simulation of mitochondrial metabolism using multi-agents system. In *4th International Conf. on Autonomous Agents and Multiagent Systems, W. MAS*BIOMED*, Pays-Bas.
- Langtangen, H. P. (2003). *Computational partial differential equations. Numerical methods and Diffpack programming.*, volume 1 of *Texts in computational science and engineering*. Springer, 2ème edition.
- Le Moigne, J. L. (1999). *La modélisation des systèmes complexes*. Dunod.
- Le Novere, N. and Shimizu, T. (2001). Stochsim : Modeling of stochastic biomolecular processes. *Bioinformatics*, 17 :575–576.
- Lecointre, G. and Le Guyader, H. (2001). *La classification phylogénétique du vivant*. Ed. Belin.
- Ledoux, H. and Gold, C. M. (2006). La modélisation de données océanographiques à l'aide du diagramme de voronoï tridimensionnel. *Revue internationale de géomatique*, 16(1) :51–70.
- Lloyd, C. M., Halstead, M. D. B., and F., N. P. (2004). CellML : its future, present and past. *Prog Biophys Mol Biol*, 85(2-3) :433–50.
- Loew, L. M. and Schaff, J. C. (2001). The Virtual Cell : a software environment for computational cell biology. *Trends in biotechnology*, 19(10) :401–406.
- Lytton, W. W. and Hines, M. L. (2005). Independent variable time-step integration of individual neurons for network simulations. *Neural computation*, 17(4) :903–921.
- Mansury, Y. and Kimura, M., Lobo, J., , and Deisboeck, T. (2002). Emerging pattern in tumor systems : simulating the complex dynamics of multicellular cluster structures by using a novel agent-based spatio-temporal agglomeration model. *J. theor. Biol.*, 219 :343–370.
- Mazliak, P. (2006). *La biologie au siècle des lumières. Comment « l'histoire naturelle » est devenue biologie*. Vuibert.
- Mellet d'Huart, D. (2004). *De l'intention à l'attention. Contributions à une démarche de conception d'environnement virtuel pour apprendre à partir d'un modèle de l'(en)action*. PhD thesis, Université du Maine, Le Mans.

- Mendel, G. (1866). Versuche über Pflanzen-Hybriden, (Experiments in Plant Hybridisation). In *Verhandlungen des naturforschenden Vereines in Brünn, (Proceedings of the Natural History Society of Brünn)*.
- Mendes, P. (1993). GEPASI : a software package for modelling the dynamics, steady states and control of biochemical and other systems. *Computer applications in the biosciences*, 9(5) :563–571.
- Mendes, P. (1997). Biochemistry by numbers : simulation of biochemical pathways with gepasi 3. *Trends in biochemical sciences*, 22(9) :361–363.
- Mendes, P. and Kell, D. B. (1998). Non-linear optimization of biochemical pathways : applications to metabolic engineering and parameter estimation. *Bioinformatics*, 14(10) :869–883.
- Mendes, P. and Kell, D. B. (2001). MEG (Model Extender for Gepasi) : a program for the modelling of complex, heterogeneous, cellular systems. *Bioinformatics*, 17(3) :288–289.
- Meseure, P., Davanne, J., Hilde, L., Lenoir, J., France, L., Triquet, F., and Chaillou, C. (2003). A physically-based virtual environment dedicated to surgical simulation. In *In Surgery Simulation and Soft Tissue Modeling (IS4TM'03), LNCS*, pages 38–47.
- Meseure, P. and Kheddar, A. (2006). Détection de collisions. In Fuchs, P., Moreau, G., and Tisseau, J., editors, *Le traité de la Réalité Virtuelle*, volume 3, pages 157–184. Les Presses de l'Ecole des Mines, third edition.
- Michaelis, L. and Menten, M. (1913). Die kinetik der invertinwirkung. *Biochemische Zeitschrift*, 49 :333–369.
- Milner, R. (1980). *A Calculus of communicating systems*. Springer Verlag.
- Milner, R. and Parrow, J. and Walker, D. (1992). A calculus of mobile processes, 1 2. *Information and Computation*, 100(1) :1–77.
- Moraru, I. I., Schaff, J. C., Slepchenko, B. M., and Loew, L. M. (2002). The Virtual Cell : an integrated modeling environment for experimental and computational cell biology. *Annals of the new york academy of sciences*, 971 :595–596.
- Nagasaki, M., Doi, A., Matsuno, H., and Miyano, S. (2003). Genomic Object Net : I. A platform for modelling and simulating biopathways. *Appl Bioinformatics*, 2(3) :181–4.
- Needleman, S. and Wunsch, C. (1970). A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J Mol Biol*, 48(3) :443–453.
- Okabe, A., Boots, B., Sugihara, K., and Chiu, S. N. (2000). *Spatial tessellations : Concepts and applications of Voronoi diagrams*. Probability and Statistics. Wiley, NYC, 2nd edition. 671 pages.
- Oprea, M. and Forrest, S. (1998). Simulated evolution of antibody gene libraries under pathogen selection. In *SMC'98, IEEE International Conference on Systems, Man and Cybernetics*, San Diego, CA (USA).
- Parenthoën, M. and Tisseau, J. (2005). *Tutorial book of virtual concept*, chapter Enactive modeling. ENSIAME-LAMIH/LIPSI-ESTIA, Biarritz, France, coutellier, d. and fischer, x. edition.
- Pasyk, K., Thomas, S., Hasset, C., Cherry, G., Phil, D., and Faller, R. (1989). Regional differences in capillary density of the normal human dermis. *Plastic and Reconstructive Surgery.*, 83(6) :939–45.
- Patankar, S. V. (1980). *Numerical Heat Transfer and Fluid Flow*. Hemisphere, Washington, D.C.
- Pearson, W. and Lipman, D. (1988). Improved tools for biological sequence comparison. *Proc Natl Acad Sci U S A*, 85(8) :2444–8.
- Pereira, U. (2005). Mémoire - le prurit -. Technical report, CHU, Brest (France).
- Petri, C. A. (1962). *Kommunikation mit Automaten*. PhD thesis, Bonn : Institut für Instrumentelle Mathematik, Schriften des IIM Nr. 2. Second Edition :, New York : Griffiss Air Force Base, Technical Report RADC-TR-65–377, Vol.1, 1966, Pages : Suppl. 1, English translation.
- Petrovsky, L. (1988). *Théorie des équations différentielles ordinaires et des équations intégrales*. Mir Moscou.

- Prigogine, I. and Stengers, I. (1997). *Les lois du chaos*. Flammarion.
- Querrec, G. (2005). *Simulation des phénomènes complexes en oncologie : application aux Myélomes Multiples*. PhD thesis, Université de Bretagne Occidentale.
- Querrec, R. (4 octobre 2002). *Les Systèmes Multi-Agents pour les Environnements Virtuels de Formation*. Thèse, Université de Bretagne Occidentale, Brest, <http://www.enib.fr/~querrec>.
- Ramsey, S., Orrell, D., and Bolouri, H. (2005). Dizzy : stochastic simulation of large-scale genetic regulatory networks. *Journal of bioinformatics and computational biology*, 3(2) :415–436.
- Reddy, V., M.L., M., and Liebman, M. (1993). Petri net representations in metabolic pathways. *Proc Int Conf Intell Syst Mol Biol*, 1(NIL) :328–36.
- Redou, P., Kerdélo, S., Le Gal, C. and Querrec, G., Rodin, V., Abgrall, J., and Tisseau, J. (2005). Reaction-agents : First mathematical validation of a multi-agent system for dynamical biochemical kinetics. *Lecture Notes in Computer Science*, 3808 :156–166.
- Regev, A. and Shapiro, A. (2004). *Modelling in Molecular Biology*, chapter The pi-calculus as an abstraction for biomolecular systems., pages 219–266. Springer.
- Regev, A., Silverman, W., and Shapiro, E. (2001). Representation and simulation of biochemical processes using the pi-calculus process algebra. *Pac Symp Biocomput*, NIL(NIL) :459–70.
- Reichhardt, T. (1999). It's sink or swim as a tidal wave of data approaches. *Nature*, 399 :517–520.
- Robert, F. (1995). *Les Systèmes dynamiques discrets*, volume 19. Springer, mathématiques applications edition.
- Rodin, V. (2004). *Contribution à l'utilisation de l'informatique en biologie*. PhD thesis, Habilitation à Diriger des Recherches, spécialité Informatique, Université de Rennes I, Brest (France).
- Rojdestvenski, I. (2003). VRML metabolic network visualizer. *Comput Biol Med*, 33(2) :169–82.
- Roux-Rouquié, M., Caritey, N., Gaubert, L., and C., R.-S. (2004). Using the unified modelling language (uml) to guide the systemic description of biological processes and systems. *Biosystems*, 75(1-3) :3–14.
- Santos, E. E., Guo, D., Jr., E. S., and Onesty, W. (2004). A multi-agent system environment for modelling cell and tissue biology. In *PDPTA*, pages 3–9.
- Sauro, H. M. (1993). SCAMP : a general-purpose simulator and metabolic control analysis program. *Computer applications in the biosciences*, 9(4) :441–450.
- Sauro, H. M. (2000). Jarnac : a system for interactive metabolic analysis. In Hofmeyr, J. H. S., Rohwer, J. M., and Snoep, J. L., editors, *Animating the cellular map. 9th International BioThermoKinetics Meeting*, pages 221–228. Stellenbosch University Press.
- Sauro, H. M. and Fell, D. A. (1991). SCAMP : a metabolic simulator and control analysis program. *Mathematical and computer modelling*, 15(12) :15–28.
- Schaff, J., Fink, C. C., Slepchenko, B., Carson, J. H., and Loew, L. M. (1997). A general computational framework for modeling cellular structure and function. *Biophysical journal*, 73(3) :1135–1146.
- Schaff, J. C., Slepchenko, B. M., Choi, Y. S., Wagner, J., Resasco, D., and Loew, L. M. (2001). Analysis of nonlinear dynamics on arbitrary geometries with the Virtual Cell. *Chaos*, 11(1) :115–131.
- Schaff, J. C., Slepchenko, B. M., and Loew, L. M. (2000). Numerical computer methods, part C. In Johnson, M. L. and Brand, L., editors, *Methods in enzymology*, volume 321, chapter Physiological modeling with Virtual Cell framework, pages 1–23. Elsevier.
- Schrödinger, E. (1944). *What is Life ?* Cambridge University Press.
- Schrödinger, E. (1993). *Qu'est ce que la vie*. Points/Poche.

-
- Seck, Y. (2003). *RIO : Rôles, Interactions et Organisation*. PhD thesis, Université des Sciences et Technologies de Lille.
- Slepchenko, B., Schaff, J., Carson, J., and Loew, L. (2002). Computational cell biology : spatiotemporal simulation of cellular events. *Annu. Rev. Biophys. Biomol. Struct.*, 31 :423–441.
- Slepchenko, B. M., Schaff, J. C., Macara, I., and Loew, L. M. (2003). Quantitative cell biology with the Virtual Cell. *Trends in cell biology*, 13(11) :570–576.
- Smith, T. and Waterman, M. (1981). Identification of common molecular subsequences. *J Mol Biol*, 147(1) :195–7.
- Snoep, J., Bruggeman, F., B.G., O., and Westerhoff, H. (2006). Towards building the silicon cell : A modular approach. *Biosystems*, 83(2-3) :207–216.
- Sorger, P. K. (2005). A reductionist’s systems biology. *Current Opinion in Cell Biology*, pages 9–11.
- Soula, H., Robardet, C., Perrin, F., Gripon, S., Beslon, G., and Gandrillon, O. (2005). Modeling the emergence of multi-protein dynamic structures by principles of self-organization through the use of 3DSpi, a multi-agent-based software. *BMC Bioinformatics*, 6(NIL) :228.
- Stein, L. (2004). End of the beginning. *Nature*, 431 :915–916.
- Stromback, L. and Lambrix, P. (2005). Representations of molecular pathways : an evaluation of SBML, PSI MI and BioPAX. *Bioinformatics*, 21(24) :4401–7.
- Ständer, S., Steinhoff, M., Schmelz, M., Weisshaar, E., Metze, D., and Luger, T. (2003). Neurophysiology of pruritus. *Arch Dermatol*, 139(1) :1463 – 1470.
- Subramanian, R. and Marsic, I. (2001). Vibe : virtual biology experiments. In *World Wide Web*, pages 316–325.
- Takahashi, K., Ishikawa, N., Sadamoto, Y., Sasamoto, H., Ohta, S., Shiozawa, A., Miyoshi, F., Naito, Y., Nakayama, Y., and Tomita, M. (2003). E-Cell 2 : multi-platform E-Cell simulation system. *Bioinformatics*, 19(13) :1727–1729.
- Takahashi, K., Kaizu, K., Hu, B., and Tomita, M. (2004). A multi-algorithm, multi-timescale method for cell simulation. *Bioinformatics*, 20(4) :538–546.
- Takahashi, K., Yugi, K., Hashimoto, K., Yamada, Y., Pickett, C. J., and Tomita, M. (2002). Computational challenges in cell simulation : A software engineering approach. *IEEE - Intelligent System*.
- Takahashi, K. and Arjunan, S. and Tomita, M. (2005). Space in systems biology of signaling pathways - towards intracellular molecular crowding in silico. *FEBS*.
- Thieffry, D. and De Jong, H. (2002). Modélisation, analyse et simulation des réseaux génétiques. *Médecine/Science*, 18 :492–502.
- Tisseau, J. (2001). *Réalité Virtuelle – autonomie in virtuo –*. PhD thesis, Habilitation à Diriger des Recherches, spécialité Informatique, Université de Rennes I, Rennes (France).
- Tisseau, J. and Harrouet, F. (2003). *Le traité de la réalité virtuelle*, volume 2, chapter 2, pages 85–119. Les presses de l’Ecole des Mines.
- Tomita, M., Hashimoto, K., Takahashi, K., Shimizu, T. S., Matsuzaki, Y., Miyoshi, F., Saito, K., Tanida, S., Yugi, K., Venter, J. C., and Hutchison, C. A. (1999). E-CELL : software environment for whole-cell simulation. *Bioinformatics*, 15(1) :72–84.
- Toyoda, M., Makino, T., Kagoura, M., and Morohashi, M. (2000). Immunolocalization of substance p in human skin mast cells. *Arch Dermatol Res*, 292(8) :418–21.

- Troncale, S., Tahı, F., Campard, D., Vannier, J. P., and J., G. (2006). Modeling and simulation with hybrid functional petri nets of the role of interleukin-6 in human early haematopoiesis. In *Proceedings of The Eleventh Pacific Symposium on Biocomputing (PSB)*.
- Varela, F. (1989a). *Autonomie et connaissance. Essai sur le vivant*. Editions du Seuil.
- Varela, F., Thompson, E., Rosh, E., and Havelange, V. (1993). *L'inscription corporelle de l'esprit. Sciences cognitives et expérience humaine*. Seuil, Paris.
- Varela, F. J. (1989b). *Invitation aux sciences cognitives*. Éditions du Seuil.
- Vervloet, D. and Magnan, A. (2003). *Traité d'allergologie*. Médecine-Science - Flammarion.
- Walker, D. C., S., S. J., Hill, G., Holcombe, M., R., H. D., Wood S. M. MacNeil, S., and Smallwood, R. H. (2004). The epitheliome : modelling the social behaviour of cells. *BioSystems*, (76) :89–100.
- Watson, J. and Crick, F. (1953). A structure for desoxyribose nucleic acid. *Nature*, 171 :737–738.
- Webb, K. and White, T. (2004). Cell modeling using agent-based formalism. In *Autonomous Agent and Multi Agent System AAMAS*, New York, USA.
- Wiener, N. (1952). *Cybernétique et société*. Éditions des Deux-Rives.
- Wiggins, S. (1990). *Local Bifurcations*, chapter Introduction to applied non linear dynamical systems and chaos, pages 25–31. Springer-Verlay, New York.
- Wishart, D., Yang, R., Arndt, D., and Cruz, J. (2004). Dynamic cellular automata : an alternative approach to cellular simulation. In *Silico Biology*, 5.
- Wooldridge, M. and Jennings, N. (1995). Intelligent agents : Theory and practice. *The Knowledge Engineering Review (Cambridge University Press)*, 10(2) :115–152.
- Yang, Y., Engin, L., Wurtele, E. S., Cruz-Neira, C., and Dickerson, J. A. (2005). Integration of metabolic networks and gene expression in virtual reality. *Bioinformatics*, 21(18) :3645–50.
- You, L., Hoonlor, A., and Yin, J. (2003). Modeling biological systems using Dynetica—a simulator of dynamic networks. *Bioinformatics*, 19(3) :435–436.
- Zienkiewicz, O. C. and Taylor, R. L. (1989). *The Finite Element Method*. McGraw–Hill, New York, NY.

Réification des interactions pour l'expérience *in virtuo* de systèmes biologiques multi-modèles.

Résumé

Cette thèse s'inscrit dans le cadre de la modélisation de phénomènes biologiques pour leur expérimentation à travers un système de réalité virtuelle. Son objet est la définition d'un cadre générique de modélisation et d'implémentation adapté à l'étude des systèmes physiologiques.

En premier lieu, le modèle générique proposé s'appuie sur le principe de la réification des interactions en objets actifs autonomes. Ensuite, il permet l'organisation des modèles biologiques en un agencement de systèmes autonomes. Il rassemble alors deux conceptions de l'autonomie : l'une est destinée à concevoir les systèmes de réalité virtuelle et l'autre a pour objet la modélisation en biologie.

Le modèle générique est dérivé en un certain nombre d'outils de modélisation pour la biologie. La bibliothèque composée du modèle générique et des outils de modélisation permet alors de réaliser différentes applications. La principale application a pour objet la mise en œuvre d'un modèle du phénomène d'urticaire allergique. Enfin, un modèle de système autopoïétique minimal est proposé pour illustrer les possibilités de la méthode.

Mots clés

Simulation, multi-agents, réalité virtuelle, *in virtuo*, biologie intégrative, interaction.

Interactions reification to experience biological multi-models systems *in virtuo*.

Abstract

To model biological systems and experiment them through a virtual reality application is the purpose of this thesis. The aim is to provide the definition of a generic modelling framework and its implementation for the study of physiological systems.

In the first place, the generic model is based on the reification of interactions into autonomous active objects. Thereby, the biological models can be organized in a layout of autonomous systems. Therefore, the generic model infers two conceptions of autonomy : the first one is used to design virtual reality systems and the second one is oriented towards biological modelling.

The generic model is specialized into several modelling tools for biology. Thereafter, the library composed by the generic models and the tools allows the building of applications. The purpose of the main application is to implement the model of an allergic urticaria phenomenon. At last, the model of a minimal autopoietic system exemplifies the method's potentials.

Keywords

Simulation, multi agent, Virtual reality, *in virtuo*, systems biology, interaction.