
Rapport scientifique sur l'ACI JC9077 « NANOPROG : une approche langage pour le nanocalcul et la simulation des nanosystèmes biologiques. »

MICHEL Olivier,	lab. IBISC,	FRE 3190,	Université d'Évry	porteur du projet
DELAPLACE Franck,	lab. IBISC,	FRE 3190,	Université d'Évry	
GRUAU Frédéric,	lab. LRI,	UMR 8623,	Université d'Orsay	
COHEN Julien,	lab. LINA,	UMR 6241,	Université de Nantes	
SPICHER Antoine,	lab. LORIA,	Projet MAIA,	INRIA Lorraine	

Table des matières

1	Le projet NANOPROG proposé	1
1.1	Objectifs scientifiques	1
1.2	Retombées attendues	1
1.3	Contexte scientifique	2
1.4	Approches et moyens scientifiques mis en œuvre	4
1.5	Collaborations	7
2	Résultats obtenus	9
2.1	Travaux théoriques	10
2.2	Validations des développements proposés	12
2.3	Diffusion de la connaissance	13
3	Perspectives futures	17
3.1	Du nano aux grands réseaux	17
3.2	Directions de recherches	18
4	Publications des membres de l'ACI	21
4.1	Journaux internationaux	21
4.2	Édition	22
4.3	Chapitres de livres	22
4.4	Publications dans des conférences internationales (avec comité de sélection)	22
4.5	Publications dans des conférences nationales (avec sélections)	23
4.6	Rapports de recherche et rapports de contrats	23
4.7	Autres publications	23
4.8	Thèses	24
4.9	Mémoire de 3 ^e cycle et de thèses	24
5	Bibliographie du document	27

Chapitre 1

Le projet NANOPROG proposé

Nous rappelons dans ce chapitre les termes du projet proposé à l'ANR.

1.1 Objectifs scientifiques

Le projet de recherche proposé a pour objectif de concevoir et d'étudier des mécanismes de programmation et des structures de données adaptés à la modélisation, la *simulation*, la *conception* et l'*exploitation* des nouvelles ressources de calcul proposées par les nanosciences et la biologie moléculaire.

Le développement d'une maquette permettant d'évaluer les mécanismes proposés est planifié sur 3 ans. Les notions développées seront mises en œuvre et validées dans trois domaines :

- la conception, l'étude et la simulation d'un *matériau intelligent programmable* (ou UPIM pour « universally programmable intelligent matter ») ;
- le développement d'une nouvelle approche pour la modélisation des systèmes à structure dynamique ;
- les applications proposées par nos partenaires (CIRAD Montpellier, Université de Calgary - canada) dans le domaine de la modélisation de processus de développement.

1.2 Retombées attendues

Les applications visées sont : le développement d'un langage adapté à la simulation des nanomatériaux et du comportement collectif des biomolécules, des outils informatiques pour la modélisation et la simulation centrée-individu du développement biologique à une échelle cellulaire, une plate-forme pour la conception et le développement de nouveaux algorithmes non déterministes et robustes aux défaillances du support de calcul (par exemple par auto-stabilisation). Plus généralement, un support langage pour toutes les applications reliées aux domaines de l'*amorphous computing* [AAC⁺00, Mac03] du *chemical computing* [DZB01, Dit03] de l'*autonomic computing* [Hor01, PH03] et des *self-healing systems* [ACM02] (systèmes distribués auto-réparateurs, calcul robuste, programmation générique, systèmes évolutifs et adaptables).

1.3 Contexte scientifique

De nouvelles ressources pour le calcul. Les avancées des nanosciences et de la biologie moléculaire mettent à la disposition de l'informaticien de nouvelles ressources de calcul. Ces nouveaux supports possibles pour le calcul se caractérisent par un très grand nombre d'entités élémentaires qui interagissent et coopèrent de manière aléatoire, irrégulière, défaillante et dynamique dans le temps. Il est néanmoins possible d'obtenir un comportement global émergent et robuste qui peut servir de calcul.

De nouveaux besoins. Parallèlement à l'élaboration de ces nouveaux media de calcul, de nouveaux besoins apparaissent au niveau du matériel, du logiciel et des applications. Tout d'abord, le modèle classique séquentiel de Von Neumann atteint ses limites intrinsèques (la croissance de la fréquence des processeurs subit un coup d'arrêt) et peine à utiliser le milliard de transistors qui sera disponible d'ici 2007 (un cœur de « processeur Von Neumann » peut s'implanter avec seulement 2250 transistors). Au niveau de la production des logiciels, la programmation impérative, fonctionnelle, logique ou objet n'a pas tenu les espérances des années 80 et 90 en termes de réutilisation, de modularité, de correction, d'expressivité, d'évolutivité ou encore de robustesse [MBFG04]. Enfin, au niveau des applications, un programme est de moins en moins concevable comme une entité monolithique conçue, produite et finalisée avant d'être utilisée. Un programme est à présent une entité vue comme un cadre ouvert, qui doit intégrer dynamiquement de nouveaux services et doit faire face à la prolifération des environnements matériels et logiciels.

De nouveaux modèles de calcul. Pour utiliser ces nouvelles ressources de calcul et faire face à ces nouveaux besoins, de nouveaux modèles de calcul se développent : *chemical computing*, *biocomputing*, *DNA computing*, *aqueous computing*, *P system*, *L-system*, *splicing system*... pour n'en citer que quelques uns.

Ces nouveaux formalismes permettent de fixer le cadre théorique et les limites de ces nouveaux paradigmes. Mais ils ne permettent pas, ou donnent très peu d'informations sur les applications et le type de problèmes que pose la mise en œuvre de ces nouvelles ressources. Cette affirmation n'est pas surprenante : que l'on pense à la distance entre la machine de Turing théorique qui abstrait l'architecture séquentielle de Von Neumann, et la multitude de langages de programmation développés pour l'utilisation effective de nos ordinateurs, depuis l'assembleur jusqu'aux langages à objets, logiques ou fonctionnels.

Les langages de programmation permettant d'observer, de contrôler, d'organiser et d'exploiter les comportements d'une collection d'entités élémentaires spatialement localisées et dont le comportement et la structure varient dans le le temps, sont donc encore à inventer.

1.3.1 Contexte international

L'analyse précédente est largement reprise de plusieurs projets américains, en particulier le projet *amorphous computing* du MIT qui a débuté dès 1998, la mouvance *autonomic computing* mise en avant par IBM, ainsi que les différents projets lancés ces dernières années autour des nanotechnologies : voir par exemple le site <http://nano.gov>.

Si les recherches et le développement de technologies permettant de manipuler des entités à l'échelle moléculaire ou macro-moléculaire sont aussi reconnus comme un domaine stratégique en Europe (cf. par exemple les initiatives lancées par l'ETI¹ et NEST²) et en France (cf. par exemple le développement du pôle Minatech de Grenoble), les conséquences pour la programmation sont à peu près ignorées, à l'exception notable de la chimie artificielle, une approche surtout étudiée en Allemagne. Les recherches européennes existantes sur les nouveaux modèles de calcul³ se cantonnent presque exclusivement sur les conséquences théoriques (calculabilité et hiérarchie de complexité) et ne considèrent ni les nouveaux domaines d'application, ni les conséquences sur les langages de programmation.

Cette constatation a motivé la co-organisation⁴ d'un workshop international [MBFG04], financé par la commission européenne (mais aussi par la NSF), afin de permettre aux différents chercheurs en programmation intéressés par ces nouvelles ouvertures de se rencontrer et de développer des collaborations.

Le projet que nous proposons ici s'inscrit dans cette perspective et vise à rattraper le retard pris en France dans ce domaine. Notons que le terme de *biotique* a été proposé dès 1981 par J. De Rosnay pour désigner le mariage de la biologie et de l'informatique dans des matériaux intelligents.

1.3.2 Insertion dans le projet scientifique du laboratoire IBISC

L'application de méthodes spécifiquement informatiques à la modélisation et à la simulation des systèmes biologiques est un des thèmes centraux du projet scientifique d'IBISC⁵. Chaque équipe mène une double recherche en concevant et développant des thématiques ancrées en « informatique classique » (génie logiciel, graphe et optimisation, architecture matérielle, système concurrent et multi-agent) dont les concepts et outils sont ensuite appliqués à des problèmes de post-génomique (analyse des réseaux de régulation, ontologie des objets et des processus biologiques au niveau moléculaire, cellulaire et tissulaire, modélisation et simulation de la cellule, déformation des membranes, etc.).

Le projet NANOPROG instancie parfaitement cette démarche, à partir de notions et de techniques déjà étudiées dans l'équipe SPECIF (méthodes formelles et programmation déclarative). Un des domaines d'application choisi pour valider le programme de recherche proposé est la simulation du développement du dôme apical chez une plante, projet mené par l'équipe de C. Godin au CIRAD et de I. Traas à l'INRA, avec lesquels nous travaillons déjà. Par ailleurs, IBISC collabore aussi, *via* un PPF, avec l'IRCOM-SIC (Poitiers, P. Lienhardt) pour ce qui est des outils de modélisation topologique, outils dont nous aurons besoin.

Plus largement, le projet NANOPROG bénéficie de l'environnement offert par le programme d'Epigénomique. Ce programme, piloté par G. Bernot d'IBISC et F. Képès de Genopole, a pour objectif de développer la modélisation des systèmes biologiques complexes.

¹“Economic and Technological Intelligence Action”

²“New and Emerging Science and Technology”

³Cf. par exemple le réseau d'excellence européen MolCoNet : <http://www.disco.unimib.it/molconet/>.

⁴conjointement avec J.-P. Banâtre, P. Fradet et J.-L. Giavitto.

⁵Le laboratoire IBISC, FRE 3190, est issu de la réunion des laboratoires LaMI, UMR 8042 et LSC, FRE 2494 en Janvier 2006.

1.3.3 Collaborations pluridisciplinaires

IBISC est un laboratoire affilié à Genopole-Evry. Dans ce contexte, IBISC a co-organisé depuis trois ans maintenant des ateliers nationaux et des écoles⁶ permettant à des biologistes, des informaticiens, des mathématiciens et des physiciens de se rencontrer autour des problèmes posés par la modélisation en post-génomique. C'est dans ce cadre que nous avons noué des contacts avec C. Godin (modélisation du développement végétal au CIRAD) et V. Norris (hyperstructure, Université de Rouen). Nos relations avec ces deux chercheurs sont décrites ci-dessous.

Par ailleurs, nous collaborons depuis plus de trois ans avec l'équipe « modélisation biologique et visualisation » de P. Prunskiewicz de l'université de Calgary (canada). Cette collaboration porte sur l'extension des L systèmes et l'utilisation de la programmation déclarative pour la modélisation du développement. Notre activité conjointe dans le cadre du projet NANOPROG est décrite ci-dessous.

1.4 Approches et moyens scientifiques mis en œuvre

L'approche proposée s'appuie sur les recherches menées précédemment dans les projets **81/2** (une approche déclarative de la simulation des systèmes dynamiques) [Mic96, GMD01], **MGS** (notion de collections topologiques et extension de la réécriture à des structures non algébriques) [Gia03] ainsi que sur des notions originales développées pour la **programmation plane** (notion globale de champs, de diffusion et de front de calcul dans un automate cellulaire) [Roz87, Roz90].

L'idée principale est d'exprimer, sous une forme algébrique et à partir de leur structure spatiale, les divers mécanismes :

1. d'interaction (influence locale d'une entité sur une autre),
2. de coopération (interaction à partir d'objectifs globaux),
3. de régulation (contraintes globales sur le comportement des entités),
4. de développement (évolution de la structure du système),

qui structurent et organisent les comportements collectifs des entités programmables. L'objectif est de permettre la représentation directe de cette algèbre d'opérateurs comme des mécanismes élémentaires de calcul dans le langage de programmation. L'approche algébrique intervient de plusieurs manières : dans la formulation des problèmes, dans la spécification de l'espace de recherche des solutions [Eng90], dans l'étude des propriétés des programmes [BDM97] et dans les implantations.

Afin de tenir compte des aspects spatiaux, l'approche que nous voulons développer s'appuie sur des notions de topologie algébrique combinatoire (espace cellulaire, chemin, chaîne, cochaîne et forme différentielle discrète). Cette voie de recherche nouvelle s'est révélée très fructueuse dans le domaine de la physique computationnelle, par exemple en théorie des champs⁷, en mécanique⁸ et en physique quantique⁹.

⁶Cf. le site <http://www.ibisc.univ-evry.fr/~epigenese/>

⁷Cf. les travaux de Enzo Tonti et l'approche de la *physique discrète* [Ton74, Mat00, ea01].

⁸Cf. les travaux de Vadim Shapiro et le langage CHAIN [PS93, NU03].

⁹Cf. le programme « digital physics » initié par Edward Fredkin [Fre03] et les approches discrètes des théories de gauges, comme par exemple [Sor91, DMH94, Req98].

En informatique, l'approche topologique que nous proposons permet d'unifier la formulation de plusieurs modèles de calcul [GM01b, GM02a] comme : les automate cellulaire [TM87], Gamma et la CHAM [BB92, BFM01], les P systèmes [Pau00] les L systèmes [PL90, RS92]. L'approche topologique permet aussi de développer un traitement polytypique [Coh03] des structures de données qui va au-delà des types algébriques réguliers.

Ce programme de recherche a été initié dans le projet MGS et plusieurs succès ont déjà été obtenus, tant dans des domaines internes à l'informatique (avec par exemple le développement d'une notion de réécriture sur les tableaux multi-dimensionnels ou les graphes de Cayley [GM01a], ou encore la conception de nouveaux algorithmes de filtrage [GM02b] et de définition par cas des fonctions) que dans des applications difficiles [Gia03] (par exemple la modélisation de la croissance du meristhème apical caulinaire chez *Arabidopsis Thaliana*).

Un des grands enseignements du projet MGS est que les mécanismes développés pour faciliter la conception et le développement de simulation de systèmes complexes sont aussi des mécanismes utiles pour la programmation en général. Plusieurs algorithmes classiques ont ainsi été exprimés de manière très concise (moins d'une dizaine de lignes de programme) : divers algorithmes de tri, recherche d'un chemin hamiltonien dans un graphe, calcul de flot maximal, calcul de l'enveloppe convexe d'un ensemble de points, divers algorithmes numériques (résolution d'équations aux dérivées partielles : diffusion, ondes...), divers algorithmes d'optimisation (knapsack, maximum segment sum, recherche min-max...), etc.

Nous voulons poursuivre le travail entrepris dans le projet MGS en nous focalisant à la fois sur l'utilisation des ressources de calcul offertes par les développements des nanotechnologies et des biomolécules, et sur les difficultés à résoudre pour la modélisation et la simulation des nano- et bio-systèmes.

1.4.1 Domaine d'applications

Trois domaines d'application sont plus précisément considérés pour valider les mécanismes de programmation proposés.

Modélisation du développement du meristhème. La modélisation et la simulation du développement (en biologie) est un domaine difficile qui nécessite de coupler plusieurs descriptions hétérogènes et dynamiques : il faut par exemple coupler la dynamique de la régulation génétique avec les aspects morphologiques et mécaniques de l'organe (cellule, tissu, individu...) en développement. Les difficultés informatiques sont considérables : comment intégrer les différents modèles, pallier des défauts de connaissances, prendre en compte la synchronisation entre les différents événements biologiques..., et cela à plusieurs échelles de temps et d'espace. Mais les enjeux sont importants et les processus biologiques en jeu sont au cœur de la relation structure-fonction. Or la simulation est le premier outil permettant, dans ce domaine, de postuler et de tester des hypothèses et d'identifier les mécanismes en jeu. On peut citer par exemple l'hypothèse des champs d'inhibition ou bien les mécanismes de diffusion/inhibition/réaction permettant à la cellule d'acquérir une information de position, hypothèses qui ne peuvent être directement vérifiées expérimentalement.

Une collaboration est en cours avec avec C. Godin du CIRAD (équipe associée à l'INRIA) pour concevoir des mécanismes de programmation permettant la modélisation et la simulation de la croissance du meristème apical caulinaire chez *Arabidopsis thaliana*. Les outils informa-

tiques sont actuellement utilisés par P. Barbier de Reuille et nos propositions devraient se finaliser au courant 2005.

Matériau intelligent programmable. Un matériau intelligent est un medium sensible, adaptatif et évolutif qui possède des fonctions inscrites dans sa forme et sa matière. Par exemple, un tel matériau est capable de modifier spontanément ses propriétés physiques (forme, connectivité, couleur, résistance, viscoélasticité...) en réponse à des excitations internes ou externes. Ces fonctions lui permettent de se comporter comme un capteur, un actionneur ou comme un processeur (traiter, comparer et stocker des informations).

Un matériau intelligent peut être solide (par exemple les matériaux magnétostrictifs ou piezo-électriques), liquide (fluides électrorhéologiques, biomolécules en milieu aqueux) ou gazeux. Un matériau intelligent *programmable* est constitué d'un petit répertoire de constructions moléculaires qui peuvent être réarrangées afin de réaliser un calcul. Un tel répertoire est *universel* s'il est Turing-complet. La création d'un répertoire universel présente de grands avantages technologiques : comme un calculateur universel, ce matériau peut se « programmer » pour accomplir une grande variété de tâches.

L'étude formelle d'un matériau intelligent programmable correspond à l'étude des propriétés d'une certaine algèbre combinatoire. La logique combinatoire [CHS72] est un exemple d'une telle algèbre, étudiée en programmation, et qui offre des exemples de familles universelles.

L'objectif visé ici est d'utiliser les structures de données et les mécanismes de programmation proposés afin de simplifier l'étude et la simulation de ces matériaux. La simulation intervient à deux titres. À un niveau abstrait, il faut valider les mécanismes programmables offerts par ces matériaux. Cet objectif est similaire à ceux du projet UPIM [Mac02a, Mac02b]. À un niveau plus fin, il faut modéliser l'assemblage des ces structures supramoléculaires afin de valider l'ingénierie de ces matériaux. L'auto-assemblage est un sujet en plein essor mais les études sont soit très théoriques [WER01], soit relèvent de l'expérience de laboratoire [RLS⁺04]. Nous voulons ici développer un outil de simulation permettant de faire le pont entre les deux.

Modélisation de systèmes à structure dynamique. Un *système dynamique à structure dynamique* [GM03, Gia03] est un système qui se caractérise non seulement par sa trajectoire (la suite de ses états) mais aussi par des évolutions *structurelles*. Ce type de système est courant dans les modèles de croissance des plantes et plus généralement en biologie du développement, dans les modèles cellulaires intégrant plusieurs échelles, dans les mécanismes de transport de protéines et de compartimentalisation, etc. Mais on peut citer d'autres domaines comme la modélisation des réseaux mobiles, d'Internet et du web, le développement des villes, les embouteillages, les processus d'auto-assemblage, les réseaux auto-catalytiques en chimie, les réseaux sémantiques au cours d'un apprentissage, les comportements sociaux, etc.

L'approche topologique que nous voulons développer permet d'explicitier et de formaliser la structure des interactions entre les différents sous-systèmes [SMG04a] et par là, de les spécifier. Nous voulons appliquer les outils NANOPROG à la simulation de BLOB [GM02c, GLRT04, Gru04] et à l'étude de l'organisation de réseaux auto-référentiels, dans la lignée des travaux de Fontana et Buss [Fon92, FB94, FB96].

1.5 Collaborations

1. Participation à l'Action Spécifique *Topologie Algébrique pour l'étude des structures de calcul, et notamment de la concurrence* (Eric Goubault).
2. *Laboratoire de modélisation AMAP du CIRAD de Montpellier* : travail avec C. Godin (CIRAD–INRIA) autour de la définition d'un langage de programmation adapté à la modélisation des processus de développement cellulaire (en biologie végétale). Le langage MGS, développé dans mon projet de recherche, est actuellement utilisé par P. Barbier de Reuille dans un projet de modélisation de la croissance cellulaire.
3. *IRCOM/SIC Poitier* : collaboration en cours (dans le cadre d'un PPF) avec l'équipe de P. Liénhardt, Y. Bertrand et les membres du projet MOKA sur l'intégration de structures de données topologiques dans un langage fonctionnel et la manipulation des G-map par réécriture. Les G-cartes classiques ont été intégrées dans MGS, mais nous nous orientons à présent vers l'intégration de chaînes de G-cartes qui permettrait de généraliser les domaines représentables et de simplifier les calculs.
4. *IRISA Rennes et INRIALPE Grenoble* : organisation conjointe avec J.-P. Banâtre (IRISA) et P. Fradet (INRIALPE) d'une conférence sur les paradigmes de programmation non-conventionnels [MBFG04].
5. *LORIA Nancy* : collaboration avec P.-E. Moreau de l'équipe PROTHEO sur le filtrage. Le compilateur MGS peut déléguer le code de filtrage à TOM, un outil générique pour le filtrage AC. Nous espérons poursuivre cette collaboration en développant de nouvelles approches pour la compilation du filtrage, en particulier pour le traitement des tableaux.
6. *Laboratoire d'informatique de l'université de Calgary, canada* : une collaboration a débuté avec l'équipe « modélisation biologique et visualisation » de P. Prunsinkiewicz sur les modèles permettant la modélisation de l'interaction et de la régulation de systèmes discrets dans les processus de développement (séjours de 2 semaines à Calgary en 2002, visite de P. Prunsinkiewicz en 2002 et 2003 à Evry) [GGMP02].
7. *Université de Liverpool, Angleterre* : travail avec G. Malcolm sur l'utilisation des systèmes de réécriture et leurs généralisations pour la modélisation en biologie à un niveau cellulaire [GMM04].

Chapitre 2

Résultats obtenus

Les travaux entrepris au cours des 4 années de l'ACI ont permis d'atteindre les objectifs scientifiques que nous nous étions fixés initialement, selon les trois directions suivantes :

1. Le développement d'une nouvelle approche pour la modélisation de systèmes dynamiques à structures dynamiques, à travers des travaux théoriques fondamentaux (et leurs implantation) :
 - (a) la définition de deux langages de motifs pour le filtrage de sous-collections topologiques, cf. section 2.1.1 ;
 - (b) l'initiation du développement d'une algèbre d'opérateurs différentiels discrets, cf. section 2.1.2 ;
 - (c) l'extension de l'algorithme de Gillespie au cas du calcul par membranes imbriquées, cf. section 2.1.3 ;
 - (d) l'intégration de la collection topologique de G-carte dans le langage MGS, cf. section 2.1.4.
2. La mise en œuvre de cette approche sur des exemples concrets validant le point de vue adopté :
 - (a) l'analyse de réseaux auto-catalytiques spatialisés, cf. section 2.2.1 ;
 - (b) la modélisation de la croissance du meristhème apical caulinaire chez *Arabidopsis thaliana*, cf. section 2.2.2 ;
 - (c) l'expression déclarative d'algorithmes de subdivisions de surface et de l'auto-assemblage en algorithmique graphique, cf. section 2.2.3 ;
 - (d) la modélisation déclarative d'un processus proche du phénomène de la neurulation dans le cadre du développement embryonnaire des métazoaires triblastiques, cf. section 2.2.4.
3. La diffusion et la propagation des travaux réalisés à travers l'organisation de groupes de travail, d'encadrement d'étudiants et de workshops :
 - (a) organisation d'un séminaire à Dagstuhl en septembre 2006, cf. section 2.3.1 ;
 - (b) la participation en 2007 à l'encadrement de la première l'équipe française de biologie synthétique pour le concours international organisé par le MIT : iGEM'08, cf. section 2.3.2 ;

- (c) l'organisation d'un workshop sur l'*Amorphous computing*, cf. section 2.3.3 ;
- (d) la mise en place d'un groupe de travail parisien sur la biologie synthétique, cf. section 2.3.4.

2.1 Travaux théoriques

2.1.1 Langages de motifs pour le filtrage de collections topologiques

Problématique. La modélisation de systèmes biologiques par des formalismes à bases de règles permet de décrire le comportement du système à travers les interactions de ses constituants. La description de ces interactions repose sur la définition d'un puissant langage de motifs.

Résultats. Les *collections topologiques* permettent de représenter un grand nombre de structures de données. Pour les manipuler, MGS fournit une structure de contrôle, appelée *transformation*, permettant la définition par cas de fonctions sur les collections topologiques. La généralité est capturée par l'utilisation de la relation de voisinage locale fournie avec chaque collection. Cette relation est le point de départ pour l'élaboration de deux langages de motifs filtrant des sous-parties d'une collection. Chaque cas de la transformation est alors une règle de réécriture ; son application correspond à une modification locale de la collection.

Une règle de réécriture d'une transformation consiste en une partie gauche et une partie droite. La partie gauche, par la spécification d'un motif de filtrage, sélectionne dans une collection topologique une sous-collection d'éléments en interaction. Selon la nature de la collection topologique et des éléments en interaction, nous avons défini deux langages de motifs : les *chemins* et les *patches*.

Ces travaux sont présentés dans les références suivantes [Spi03, SM04a, SM04b, Spi06, SM07, GSM07].

2.1.2 Algèbre d'opérateurs différentiels discrets

Problématique. Pour permettre la description et l'intégration de la notion de champs, associer des quantités à des morceaux d'espaces (concentration pour un volume, flux sur une surface...) et enfin disposer d'une unique notation quelle que soit la nature, discrète ou continue, des objets considérés, il est nécessaire de développer une algèbre d'opérateurs différentiels dans un cadre discret.

Résultats. Nous avons établi un parallèle entre la notion de *transformation* en MGS et des notions issues du calcul différentiel discret. En particulier, la formalisation des collections topologiques comme des chaînes topologiques amène naturellement vers les concepts de cochaînes et de formes différentielles discrètes. Cette analogie réalise les bases pour permettre la mise en place d'une algèbre de calcul sur les transformations et les collections topologiques, suivant une direction de recherche proche des algèbres de Bird-Merteens.

Ces travaux sont présentés dans les références suivantes [GSM07, GS08].

2.1.3 Un algorithme de simulation stochastique exact pour les membranes imbriquées

Problématique. La modélisation et la simulation de systèmes biologiques à l'échelle micro met en jeu un faible nombre de molécules (de l'ordre de la dizaine). À cette échelle, les modèles continus ne fonctionnent plus et il est alors nécessaire d'utiliser des modèles stochastiques. Pour prendre en compte des phénomènes structurés, il devient nécessaire d'étendre ces modèles en conséquence.

Résultats. La modélisation de systèmes dynamiques qui impliquent un faible nombre d'individus, comme c'est le cas dans les réactions moléculaires bio-chimiques, ne peuvent pas être décrits par des équations différentielles. En effet, les variations de densité de populations sont trop importantes et le bruit joue un rôle important qui n'est pas capturé par une modélisation continue. Pour pallier ces limitations, D.T. Gillespie [Gil77] a défini un algorithme de simulation stochastique exact permettant la modélisation exacte d'équations bio-chimiques, considérant un milieu homogène spatialement.

Nous avons proposé une extension de cet algorithme au cas où les molécules chimiques en réactions sont compartimentalisées et séparées par des membranes. L'algorithme a été proposé dans le cadre du formalisme des *P systems* et intégré dans le langage MGS. Nous avons proposé une modélisation du cycle de vie du virus *Semliki* basé sur ce nouvel algorithme.

Ces travaux sont présentés dans les références suivantes [Spi06, SMC+08].

2.1.4 Intégration de la collection topologique de G-carte

Problématique. La représentation de géométries complexes nécessite de disposer de structures de données permettant d'effectuer des opérations coûteuses avec les meilleures performances. Pour ces raisons, on s'intéresse à l'intégration de la structure de données de G-carte.

Résultats. Les collections topologiques ont été formalisées à travers des concepts issus de la topologie algébrique combinatoire et notamment la notion de *complexe cellulaire*. Un complexe cellulaire est une collection d'objets de dimensions différentes appelés *k-cell* (ou cellule de dimension k). Les 0-cells sont des sommets, les 1-cells des arcs, les 2-cells des faces, les 3-cells des volumes... Une k -cell c est incidente à une $(k - 1)$ -cell c' si $c' \subset \partial c$, où ∂c dénote le bord de c .

L'utilisation de complexes cellulaires abstraits permet de représenter un grand nombre de configurations topologiques complexes. Cependant, de par leur nature combinatoire, leur usage est très coûteux et les performances des algorithmes sont mauvaises. Pour disposer d'une implantation efficace, nous avons intégré dans MGS la structure de données de G-cartes, en collaboration avec le laboratoire IRCOM-SIC de Poitiers¹, permettant de représenter des *quasi-variétés*. Leur implantation a permis la mise en œuvre d'algorithmes classiques en CAO avec des performances raisonnables.

Ces travaux sont présentés dans les références suivantes [SMG04b, SM04b].

¹<http://www.sic.sp2mi.univ-poitiers.fr/>

2.2 Validations des développements proposés

2.2.1 Analyse de réseaux auto-catalytiques

La *chimie artificielle* est une étude par des outils théoriques, des problèmes posés par l'analyse des réactions chimiques qui ont lieu dans les organismes biologiques. Plus précisément, on s'intéresse aux systèmes auto-catalytiques qui ont la propriété de former des cycles de dépendances et sur lesquels on se pose des questions d'émergence, de stabilité, de conditions nécessaires à leur établissement... Les molécules qui apparaissent dans des cycles auto-catalytiques sont parfois qualifiées d'auto-répliquantes et l'analogie avec les systèmes biologiques est évidente. Ces questions sont proches de la notion d'autopoïèse proposée par F. Maturana et H. Varela [VMU74].

Nous avons analysé les conditions d'apparitions de cycles auto-catalytiques dans le cadre du travail de stage de DEA de N. Mann [Man04] en suivant la ligne des travaux de W. Fontana et L. Buss [FB93] qui représentent les molécules des systèmes chimiques artificiels par des termes du λ -calcul. Nous avons pour cela introduit dans MGS la capacité d'effectuer des évaluations de termes *bornées en nombre de β -réductions* et avons analysé les cycles produits. Nous avons enfin amorcé un travail de *localisation* des réactions chimiques sur des espaces homogènes, représentés par des GBFs [GM01a, GMC02]. Ces travaux demandent à être poursuivis.

2.2.2 Modélisation de la croissance du meristhème apical caulinaire

Nous avons collaborés avec l'équipe de C. Godin à l'INRA² Montpellier et de J. Traas au CNRS/INRA³ à Lyon dans le cadre de la modélisation de la croissance du meristhème apical caulinaire chez *Arabidopsis Thaliana*.

Le transport actif de l'auxine, protéine hormonale des plantes, joue un rôle majeur dans la production des organes dans le méristhème apical. La localisation de protéines membranaires de la famille PIN1 facilite le transport et de récentes observations suggèrent que la concentration élevée d'auxine est à l'origine de l'initiation d'un organe. Cette hypothèse est fondée sur l'évaluation visuelle et qualitative de la distribution complexe des motifs de la protéine PIN1 chez *Arabidopsis*. Pour aller au delà de cette évaluation informelle, les travaux initiés par l'équipe de C. Godin et de J. Traas, dans le travail de thèse de P. Barbier de Reuille ont proposé un modèle formel. La simulation de ce modèle a permis de confirmer les hypothèses quant à la distribution de la protéine suivant des motifs précis.

Grâce à une forte collaboration, le langage MGS a été utilisé (et a évolué pour prendre en compte ce domaine d'utilisation, par exemple la spécialisation de la collection topologique de graphe de Delaunay) pour la modélisation et la simulation de la couche L1 en croissance du méristème et intégré dans le simulateur *Merrysim* réalisé par P. Barbier de Reuille. Ces travaux ont donné lieu aux publications suivantes [BdR05, BdRBCL⁺06].

²http://amap.cirad.fr/fr/mia_theme1.php

³<http://www.ens-lyon.fr/web/nav/article.php?id=132>

2.2.3 Spécification déclarative d'algorithmes de subdivisions de surface et d'auto-assemblage

Dans le cadre d'une collaboration avec le laboratoire du Département d'Informatique de l'université de Calgary, le professeur P. Prusinkiewicz⁴ avait posé la question de savoir s'il était possible de décrire par des règles de modifications purement locales des algorithmes classiques de subdivisions de surfaces en CAO. En effet, la *spécification* de ces algorithmes s'exprime en termes de modifications purement locales alors que la mise en œuvre nécessite la manipulation de la structure globale. Ces implantations ne permettent alors plus d'établir des propriétés (telles que la correction ou la complexité) sur les algorithmes ainsi produits.

Nous avons montré dans [SMG05, GS06, SM07] que l'on pouvait décrire en MGS ces algorithmes selon un point de vue purement local : des règles de réécritures spécifient les opérations de subdivisions de surfaces *locales* et l'application de ces règles à la structure complète permet de *propager* les modifications à l'objet considéré.

2.2.4 Modélisation déclarative d'un processus de neurulation

La biologie du développement investigue des systèmes complexes dont l'organisation temporelle et spatiale est fondamentale. En effet, ce sont les relations spatiales entre les constituant d'un organisme en développement qui contraignent les histoires possibles de l'organisme. Dans ce cadre, nous avons montré dans [SM05, SM06] comment les principes développés dans le cadre du projet MGS pouvaient permettre la modélisation et la simulation d'un phénomène impliquant un système dynamique à structure dynamique : les premières étapes du processus de neurulation (où la crête neurale se forme à partir d'un feuillet de cellules). L'usage conjoint de la collection topologique de complexe cellulaire et des règles de transformations a permis une première expression concise de cette étape du développement.

2.3 Diffusion de la connaissance

Les travaux présentés ici représentent une partie des travaux qui ont été effectués dans le cadre de cette ACI. Conjointement aux développements scientifiques, un important travail de diffusion de la connaissance a été fait afin de présenter et proposer à la communauté scientifique les avantages liés à l'approche que nous défendons.

2.3.1 Séminaire Dagstuhl sur le « Calcul Spatial »

Nous avons organisé un séminaire du 3 au 8 septembre 2006 à Dagstuhl⁵ intitulé « Computing Media and Languages for Space-Oriented Computation » durant lequel nous avons réuni des membres de la communauté des architectes, des langages et des applications autour de la problématique de la prise en compte de l'espace dans les langages, les applications et les architectures.

⁴<http://algorithmicbotany.org/>

⁵Le site du séminaire est accessible à l'url <http://www.dagstuhl.de/en/program/calendar/semhp/?semnr=06361>

Durant ce séminaire, nous avons forgé le concept de « calcul spatial » où, dans les langages comme dans les architectures, la notion d'espace ne peut plus être ignorée. En effet, dans les nouvelles architectures où l'accès à la mémoire n'est pas uniforme selon la localisation de l'élément de calcul accédé, il est vital de rendre explicite le coût d'accès à une ressource selon sa position (physique ou logique). Il est ainsi nécessaire de rendre disponible cette information au niveau de l'architecture mais aussi au niveau du langage qui doit permettre la prise en compte de cette information lors de la définition d'algorithmes.

À l'issue de cette réunion, nous avons entamé un certain nombre de collaborations, en particulier avec le professeur D. Coore.

2.3.2 Concours international de biologie synthétique iGEM 2007

À la suite de leurs travaux sur le « Calcul amorphe », les informaticiens et les biologistes du MIT ont proposé de changer le point de vue classique sur la biologie. Par un important effort de standardisation, de découplage et d'abstraction, des chercheurs tels que R. Weiss, D. Endy et T. Knight ont proposé un point de vue d'ingénieur sur la biologie moléculaire. Faisant suite à la *biologie systémique*, l'essor de la *biologie synthétique* est censé prendre le relais de la Loi de Moore qui, pour des raisons technologiques, arrive à son terme.

En appliquant la démarche qui a permis à l'électronique de passer du stade artisanal au stade industriel, la hiérarchisation des assemblages de gènes en *fonction*, *device* et *systèmes* permet la construction d'organismes complexes par composition modulaire. La réalisation effective se fait *via* l'usage des ressources disponibles dans le vivant (essentiellement la bactérie *Escherichia coli*) en y introduisant un plasmide portant la suite des nucléotides codant pour le « programme » à réaliser : la machinerie cellulaire de l'hôte se charge alors de *réaliser* la fonction construite.

Se basant sur ces principes, le MIT organise chaque année une compétition internationale où des étudiants en master, encadrés par des étudiants en thèse et des chercheurs confirmés, proposent, développent et réalisent une fonction biologique : c'est le concours iGEM⁶. L'effectivité de la méthode permet à ces étudiants, en environ 3 mois, de produire un travail qui aurait nécessité, dans un cadre de génie génétique standard, le travail d'un étudiant durant une thèse complète.

En 2007, nous avons participé à l'encadrement de la première équipe française au concours iGEM. Lors de cette participation, le langage MGS a été utilisé avec succès pour modéliser et simuler l'organisme synthétique multicellulaire proposé⁷. À l'issue de ce concours, l'équipe française a reçu le premier prix dans la catégorie « Recherche fondamentale ».

Cette année, en 2008, nous participons de nouveau à l'encadrement de l'équipe française pour sa seconde participation.

2.3.3 Workshop sur le Calcul Amorphe

Suite au séminaire de Dagstuhl, nous avons invité D. Coore, ancien étudiant de H. Abelson et G.J. Sussman au MIT, en juin 2007 afin d'entamer une réflexion sur la définition d'un nouveau

⁶international Genetically Engineered Machine competition.

⁷Le site de l'équipe parisienne d'iGEM est <http://parts.mit.edu/igem07/index.php/Paris/> et l'on trouve à l'url suivante les différents modèles effectués en MGS <http://parts.mit.edu/igem07/index.php/Paris/Modeling>

langage reposant sur les nouvelles ressources disponibles en biologie synthétique (cf. ci-avant) et mettant en œuvre les résultats du calcul amorphe dans un contexte de calcul spatial.

Les premiers éléments du langage ont été définis lors de la visite du professeur D. Coore et nous poursuivons actuellement, grâce au recrutement d'A. Spicher à l'université de Paris-12, les travaux entamés. Conjointement à la visite du professeur D. Coore, nous avons organisé un *workshop* autour du calcul amorphe à Paris afin de réunir pendant une journée les acteurs francophones concernés par le calcul amorphe en général. Le site est accessible à l'adresse <http://amorphous-07.ibisc.univ-evry.fr>

Cette année, en 2008, nous organisons sur deux journées la suite du précédent colloque ; nous aurons comme orateurs invités G.J. Sussman (MIT), A. Adamatzky (Université de Bristol) et C. Teuscher (LANL) . Le site est accessible à l'url <http://amorphous.ibisc.univ-evry.fr>

2.3.4 Groupe de travail parisien sur la biologie synthétique

Notre engagement en biologie synthétique, à travers notre participation au projet iGEM et l'évolution de nos travaux de recherche dans cette direction, nous ont amené à proposer un groupe de travail autour de la biologie synthétique. Ce groupe de travail qui réunit des chercheurs biologistes, physiciens et informaticiens se réunit tous les mois et se veut un lieu de réflexion et d'élaboration de nouveaux outils et objets (théoriques et pratiques) pour mettre en œuvre les ressources offertes par la biologie synthétique.

Parallèlement à ce groupe de travail, nous maintenons, avec S. Bottani du laboratoire MSC de l'université Paris 7, deux sites Web francophones sur le sujet de la biologie synthétique. Ces sites sont accessibles aux url suivants <http://sb.ibisc.univ-evry.fr/> et <http://biologiesynthetique.fr/>

Chapitre 3

Perspectives futures

Les travaux que nous avons réalisés durant cette ACI nous permettent d'envisager de nombreuses pistes originales et prometteuses. Parmi celles-ci, nous en détaillons deux directions essentielles pour comprendre comment décrire et programmer le comportement de *populations* d'entités (nano-particules, cellules biologiques, essaims robots. . .) en interaction du niveau micro (en biologie synthétique) au niveau macro (les grilles de calcul).

3.1 Du nano aux grands réseaux

3.1.1 Programmer des populations en biologie synthétique

Les avancées récentes des nanotechnologies et de la biologie nous permettent maintenant de construire des fonctions biologiques par l'assemblage de briques de base : c'est l'approche promue par la *biologie synthétique*. Cette approche permet d'aller au-delà du génie génétique standard pour dorénavant *programmer* le fonctionnement d'organismes biologiques.

Le MIT a proposé une ontologie (sous la forme d'une structuration des entités allant des nucléotides aux systèmes complets en passant par les niveaux des promoteurs, opérateurs, régions codantes, sites de liaisons des ribosomes...) et un principe de mise en œuvre : l'intégration d'un plasmide dans une bactérie (généralement *E-coli*) qui permettra la réalisation effective de la fonction programmée.

Cette approche est utilisée chaque année dans le cadre du projet universitaire iGEM (« international Genetically Engineered Machine »¹) qui met en compétition des étudiants *undergraduates* pour la définition de nouvelles entités biologiques. Ces organismes bio-synthétiques représentent essentiellement des entités autonomes qui réalisent une fonction simple de type senseur-effecteur.

Nous souhaitons aller au-delà des simples réalisations qui sont aujourd'hui proposées. En effet, il est dorénavant possible de construire des *populations* de bactéries pouvant communiquer avec leur environnement immédiat et réalisant des fonctions simples. L'utilisation coordonnée de ces populations de bactéries doivent permettre la construction d'un calcul et la réalisation d'algorithmes.

¹Le projet est décrit à l'url suivante http://parts.mit.edu/igem07/index.php/Main_Page

3.1.2 Les grands réseaux et le calcul autonome

Conjointement à l'apparition de ces nouvelles ressources de calcul à une échelle de la matière la plus petite, les réseaux d'ordinateurs classiques (*via* Internet et les applications de *peer-to-peer* par exemple) offrent, à l'autre extrémité de l'échelle, des ressources de calcul qui correspondent à un système dynamique à structures dynamiques : le comportement par défaut d'un système composé d'une grande population d'entités simples de calcul correspond à un point-fixe du système (état stable) ; une perturbation ou un changement de structure correspond à un éloignement vis-à-vis du point-fixe ; le retour du système à son état stable correspond aux propriétés d'*auto-** (-stabilisation, -correction, -organisation...).

Ces ressources partagent les mêmes caractéristiques que les entités de la biologie synthétique :

- ces ressources sont *localisées* dans l'espace et cette information est cruciale : le coût de l'accès n'est pas le même (et parfois rédhibitoire) selon que la ressource est proche ou distante (logiquement ou spatialement),
- ces ressources n'ont *pas de contrôle global* et ne peuvent être de fait décrites que localement,
- ces ressources sont potentiellement *infinies* et dynamiques : elles sont potentiellement défaillantes, les capacités de communication peuvent s'altérer,
- ces ressources sont qualifiés d'*éternelles*, c'est-à-dire qu'elles ne fonctionnent pas comme un ordinateur que l'on pourrait allumer ou éteindre ; en effet, ces ressources ont, d'une certaine façon « toujours fonctionné et fonctionneront toujours ».

Ces propriétés nécessitent d'articuler la description des entités à un niveau **local** afin d'assurer l'obtention d'un comportement au niveau **global**. Cela peut se résumer selon le principe de *spécifier un calcul localement pour le calculer globalement*. Il semble évident que de telles ressources ne pourront pas résoudre des problèmes algorithmiques classique mais plutôt maintenir un état stable comme solution d'un problème et exhiber des propriété d'*auto-**.

3.2 Directions de recherches

3.2.1 Modélisation hybride de phénomènes de diffusion-réaction

Un problème fondamental dans les simulations *centrées individus*² est que le temps de calcul augmente linéairement avec le volume simulé et toujours plus important que l'algorithme SSA de Gillespie (cf. la section 2.1.3). Par conséquent, la simulation centrée individus de systèmes dont la taille est réaliste (c'est-à-dire de dimensions physiques réalistes, avec un temps de simulation réel conséquent et un niveau de détail correct du comportement simulé) ne sont actuellement pas atteignables.

En collaboration avec H. Berry (INRIA Saclay, projet Alchemy), nous avons commencé à développer de nouvelles méthodes pour accélérer ces simulations. Ces méthodes reposent sur des schémas de simulations hybrides où l'on combine des techniques individus centrées, à l'échelle où elles sont correctes, avec des équations déterministes (c'est-à-dire basées sur les lois d'action de masse) pour toutes les autres échelles. Ces travaux vont nous permettre de proposer des

²Nous préférons cette appellation à celle de *systèmes multi-agents* car elle ne fait pas référence aux travaux explicites et aux références cognitives de ce paradigme.

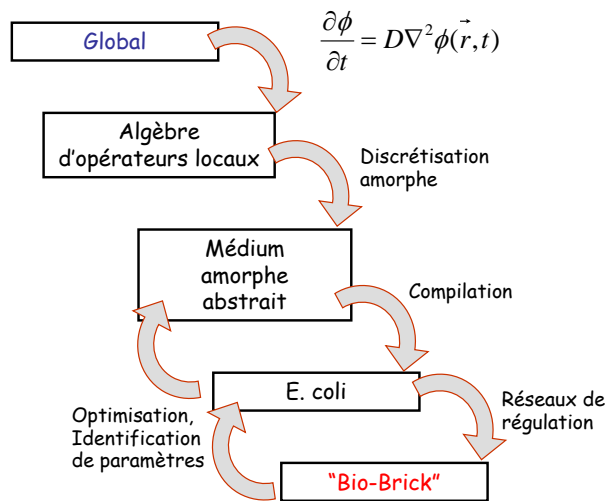


FIG. 3.1 – Schéma général d’une hiérarchie de langages pour la programmation des nouvelles entités amorphes du calcul spatial.

méthodes de modélisation et de simulation originales pour la biologie synthétique en prenant en compte la spécificité (distribution spatiale, homogénéité...) du niveau de description considéré.

3.2.2 Programmer les nouveaux médias de calcul.

À partir des résultats obtenus dans le projet MGS et dans le cadre du calcul amorphe, nous proposons de développer de nouveaux outils pour la programmation de ces grandes populations d’entités dynamiques, autonomes, possiblement défaillantes et dont les possibilités de communications sont limitées. Nous proposons la définition d’une hiérarchie de langage, avec au niveau le plus élevé des primitives permettant le *calcul spatial* : gradient, vague, champs... et au niveau le plus bas les *bio-bricks* proposées dans l’ontologie du MIT. Chaque niveau doit être affiné :

1. vers une algèbre d’opérateurs locaux correspondant à un analogue discret des opérateurs du calcul différentiel continu,
2. puis une discrétisation amorphe (en effet, les médias de calcul que l’on considère sont irréguliers, défaillants et asynchrones),
3. puis une étape de compilation vers les *instructions* d’une bactérie idéale (avec des instructions telles que la possibilité de changer sa polarisation, de sécréter des substances, de prendre/déposer de la matière, de croître, se diviser...),
4. enfin, la traduction finale en termes de *biobricks*.

Les étapes (3–4) nécessiteront un schéma de compilation itératif (présenté à la figure 3.1) afin de résoudre les problèmes de spécifications incomplètes des *biobricks* dont on dispose à l’heure actuelle.

Chapitre 4

Publications des membres de l’ACI

Dans chaque section sont présentées les publications des membres de l’ACI en rapport avec le projet. Les entrées sont par ordre chronologique puis par ordre alphabétique. Toutes les publications sont disponibles sur requête aux auteurs.

4.1 Journaux internationaux

- [J1] Jean-Louis Giavitto, Grant Malcolm, and Olivier Michel. Rewriting systems and the modelling of biological systems. *Comparative and Functional Genomics*, 5 :95–99, February 2004.
- [J2] Antoine Spicher and Olivier Michel. Declarative modeling of a neurulation-like process. *BioSystems*, 87 :281–288, February 2006.
- [J3] Olivier Michel, Jean-Pierre Banâtre, Pascal Fradet, and Jean-Louis Giavitto. Challenging questions for the rationales of non-classical programming languages. *International Journal of Unconventional Computing*, 2006.
- [J4] Antoine Spicher, Olivier Michel, Mikolaj Cieslak, Jean-Louis Giavitto, and Przemyslaw Prusinkiewicz. Stochastic P systems and the simulation of biochemical processes with dynamic compartments. *BioSystems*, 91 :458–472, March 2008.
- [J5] Jean-Louis Giavitto and Antoine Spicher. Topological Rewriting and the Geometrization of Programming. *Physica D*, 237(9) :1302–1314, July 2008.

Les quatre publications suivantes apparaissent dans un journal en langue française, *Technique et Science Informatique*.

- [J6] Jean-Louis Giavitto, Olivier Michel, Jean-Pierre Banâtre, and Pascal Fradet. Modèles de programmation non-conventionnels. *Technique et Science Informatique*, 23 :177–186, 2004. Compte-rendu de l’atelier international UPP’04. (not reviewed)
- [J7] Julien Cohen. Typage des collections topologiques hétérogènes et des transformations *Technique et Science Informatique*, 2007. 24 :1081–1111, 2005
- [J8] Julien Cohen. Interprétation par syntaxe abstraite d’ordre supérieur et traduction en combinateurs *Technique et Science Informatique*, 2007. 26 :1137–1167, 2007
- [J9] Antoine Spicher and Olivier Michel. Représentation et manipulation de structures topologiques dans un langage fonctionnel. *Technique et Science Informatique*, 2007.

4.2 Édition

Co-édition de deux actes de conférences et d'un numéro spécial dans un journal :

- [E1] Jean-Pierre Banâtre, Pascal Fradet, Jean-Louis Giavitto, and Olivier Michel, editors. *Unconventional Programming Paradigms (UPP'04)*, volume 3566 of *LNCS*, Le Mont Saint-Michel, France, September 2005. ERCIM – NFS, Springer Verlag. Revised, selected and invited papers. 367 p. ISBN : 3-540-27884-2. <http://www.springeronline.com/3-540-27884-2>.
- [E2] Olivier Michel and Pierre Weis, editors. *Seizièmes Journées Francophones des Langages Applicatifs (JFLA '05)*, number 16. INRIA, 2005. <http://jfla.inria.fr/2005/actes/actes-jfla-2005.tgz>.
- [E3] Thérèse Hardin and Olivier Michel, editors. Autour et aux alentours des langages applicatifs. *Technique et Science Informatique*, Hermès Science. 2007.

4.3 Chapitres de livres

- [B1] Jean-Louis Giavitto and Olivier Michel. *Molecular Computational Models : Unconventional Approaches*, chapter Modeling Developmental Processes in MGS, pages 1–46. Idea Group, 2004.
- [B2] Olivier Michel and Florent Jacquemard. *An Analysis of a Public-Key Protocol with Membranes*, pages 281–300. Natural Computing Series. Springer Verlag, 2005.
- [B3] Jean-Louis Giavitto and Antoine Spicher. *Systems Self-Assembly : Multidisciplinary Snapshots*, chapter Simulation of self-assembly processes using abstract reduction systems, pages 199–224. Édition N. Krasnogor, S. Gustafson, D. Pelta, J.L. Verdegay, Elsevier, 2008.
- [B4] Jean-Louis Giavitto and Antoine Spicher. *Morphogenèse*, chapitre Morphogenèse informatique, pages 178–198. Sous la direction de P. Bourguin et A. Lesne, Belin, 2006.

4.4 Publications dans des conférences internationales (avec comité de sélection)

- [IC1] Olivier Michel and Florent Jacquemard. An analysis of the needham-schroeder public-key protocol with MGS. In G. Mauri, G. Paun, and C Zandron, editors, *Preproceedings of the Fifth workshop on Membrane Computing (WMC5)*, pages 295–315. EC MolConNet - Università di Milano-Bicocca, June 2004.
- [IC2] Antoine Spicher, Olivier Michel, and Jean-Louis Giavitto. A topological framework for the specification and the simulation of discrete dynamical systems. In *Sixth International conference on Cellular Automata for Research and Industry (ACRI'04)*, volume 3305 of *Lecture Notes in Computer Sciences*, pages 238–247, LNCS, Amsterdam, October 2004.
- [IC3] Antoine Spicher and Olivier Michel. Declarative modeling of a neurulation-like process. In *Sixth International Workshop on Information Processing in Cells and Tissues*, 2005.

- [IC4] Antoine Spicher and Olivier Michel. Using rewriting techniques in the simulation of dynamical systems : Application to the modeling of sperm crawling. In *Fifth International Conference on Computational Science (ICCS'05)*, volume I, pages 820–827, 2005.
- [IC5] Jean-Louis Giavitto, Olivier Michel, and Antoine Spicher. Computation in space and space in computation. In J.-P Banâtre, P. Fradet, Jean-Louis Giavitto, and Olivier Michel, editors, *Unconventional Programming Paradigms (UPP'04)*, number LNCS 3566, pages 137–152. ERCIM– NSF, Springer Verlag, 2005.
- [IC6] Antoine Spicher, Olivier Michel, and Jean-Louis Giavitto. Algorithmic self-assembly by accretion and by carving in MGS. In *7th International Conference on Artificial Evolution*, 2005.

4.5 Publications dans des conférences nationales (avec sélections)

- [C1] Antoine Spicher and Olivier Michel. Stratégie d'application stochastique de règles de réécritures dans le langage MGS. In *Journées Francophones des Langages Applicatifs*. INRIA, 2006.
- [C2] Antoine Spicher. Représentation et manipulation de structures topologiques dans un langage fonctionnel. In *Journées Francophones des Langages Applicatifs*. INRIA, 2005.
- [C3] Julien Cohen. Interprétation par SK-traduction et syntaxe abstraite d'ordre supérieur In *Journées Francophones des Langages Applicatifs*. INRIA, 2005.
- [C4] Antoine Spicher and Olivier Michel. Manipulations de structures topologiques dans un langage déclaratif pour la simulation. In *11ème Journées du GT "Animation et Simulation" (GTAS'2004)*, Reims, juin 2004. AFIG et LERI, Université de Reims.

4.6 Rapports de recherche et rapports de contrats

- [R1] Olivier Michel, Florent Jacquemard, and Jean-Louis Giavitto. Three variations on the analysis of the needham-schroeder public-key protocol with MGS. Technical Report LaMI-98-2004, LaMI – Université d'Évry - CNRS, May 2004. 25 p.
- [R2] Antoine Spicher, Olivier Michel, and Jean-Louis Giavitto. A topological framework for the specification and the simulation of discrete dynamical systems. Technical Report LaMI-99-2004, LaMI, May 2004.
- [R3] Jean-Louis Giavitto, Antoine Spicher and Olivier Michel. Topological Rewriting and the Geometrization of Programming. Technical Report IBISC-XX-2007, IBISC, September 2007.

4.7 Autres publications

- [O1] Olivier Michel, Jean-Pierre Banâtre, Pascal Fradet, and Jean-Louis Giavitto. The Unconventional Programming Paradigms home page (UPP04). <http://upp.ibisc>.

univ-evry.fr, 2004. International workshop for “Challenges, Visions and Research Issues for New Programming Paradigms”, 15 - 17 September 2004, Mont Saint-Michel, France.

- [O2] Antoine Spicher and Olivier Michel. Integration and pattern-matching of topological structures in a functional language. In *International Workshop on Implementation and Application of Functional Languages (IFL04)*, Lübeck, September 2004.
- [O3] Olivier Michel and Jean-Louis Giavitto. Incremental extension of a domain specific language interpreter. In *International Workshop on Implementation and Application of Functional Languages (IFL07)*, Freiburg, Germany, September 2007. The draft proceedings will appear as a technical report of the Computing Laboratory of the University of Kent.

4.8 Thèses

- [T1] Olivier Michel. *There's Plenty of Room for Unconventional Programming Languages or Declarative Simulations of Dynamical Systems (with a Dynamical Structure)*. Habilitation, Université d'Évry, December 2007.

4.9 Mémoire de 3^e cycle et de thèses

- [MT1] Julien Cohen. *Intégration des collections topologiques et des transformations dans un langage fonctionnel*. Mémoire de thèse de doctorat, Université d'Évry, 2004.
- [MT2] Nicolas Mann. Hyperstructures et modélisation de chimie artificielle dans le langage MGS. Mémoire de 3^e cycle, DEA AMIB Université d'Évry, 2004.
- [MT3] Damien Boussié. Simulation en MGS du déplacement du spermatozoïde du nématode *Ascaris Suum*. Mémoire de 3^e cycle, DEA AMIB Université d'Évry, 2004.
- [MT4] Lionel Perret. Intégration des types de données algébriques dans MGS. Mémoire de 3^e cycle, École Centrale Paris, 2005.
- [MT5] Yann Jullian. Conception et développement d'un éditeur graphique de filtre pour MGS. Mémoire de 3^e cycle, IIE, 2005.
- [MT6] Fabien Gaubert. Simulation stochastique et modélisation de chimie artificielle dans le langage MGS. Mémoire de 3^e cycle, DEA AMIB Université d'Évry, 2005.
- [MT7] Antoine Spicher. *Transformation de collections topologiques de dimension arbitraires. Application à la modélisation de systèmes dynamiques*. Mémoire de thèse de doctorat, Université d'Évry, 2006.
- [MT8] Christoph Kaleta. Outils de visualisation pour la simulation de systèmes dynamiques à structure dynamique. Mémoire de 3^e cycle, Master Informatique, Universität Jena & Université d'Évry, 2007.

Bibliographie

- [AAC⁺00] Harold Abelson, Don Allen, Daniel Coore, Chris Hanson, George Homsy, Thomas F. Knight, Radhika Nagpal, Erik Rauch, Gerald Jay Sussman, and Ron Weiss. Amorphous computing. *CACM : Communications of the ACM*, 43, 2000.
- [ACM02] ACM SIGSOFT. *Workshop on Self-Healing Systems (WOSS'02)*, Charleston, USA, November 2002. ACM press.
- [BB92] Gérard Berry and Gérard Boudol. The chemical abstract machine. *Theoretical Computer Science*, 96 :217–248, 1992.
- [BDM97] Richard Bird and Oege De Moor. *Algebra of Programming*, volume 100 of *Int. Series in Computer Science*. Prentice Hall, 1997.
- [BdR05] Pierre Barbier de Reuille. *Vers un modèle dynamique du meristème apical caulinaire d'Arabidopsis thaliana*. PhD thesis, Université de Montpellier II, December 2005.
- [BdRBCL⁺06] Pierre Barbier de Reuille, Isabelle Bohn-Courseau, Karin Ljung, Halima Morin, Nicola Carraro, Christophe Godin, and Jan Traas. Computer simulations reveal properties of the cell-cell signaling network at the shoot apex in Arabidopsis. *PNAS*, 103(5) :1627–1632, 2006.
- [BFM01] Jean-Pierre Banâtre, Pascal Fradet, and Daniel Le Métayer. Gamma and the chemical reaction model : Fifteen years after. *Lecture Notes in Computer Science*, 2235 :17–44, 2001.
- [CHS72] Haskell B. Curry, J. R. Hindley, and J. P. Seldin. *Combinatory Logic*. North-Holland, Amsterdam, 1972.
- [Coh03] Julien Cohen. Typing rule-based transformations over topological collections. In Jean-Louis Giavitto and Pierre-Etienne Moreau, editors, *Electronic Notes in Theoretical Computer Science*, volume 86. Elsevier, 2003. Revised version of [?].
- [Dit03] P. Dittrich. The Artificial Chemistry Page. <http://ls11-www.cs.uni-dortmund.de/achem>, 2003.
- [DMH94] Aristophanes Dimakis and Folkert Mueller-Hoissen. Discrete differential calculus graphs, topologies and gauge theory. *J. Math. Phys.*, 35 :6703–6735, 1994.
- [DZB01] P. Dittrich, Jens Ziegler, and Wolfgang Banzhaf. Artificial chemistries - a review. *Artificial Life*, 7(3) :225–275, 2001.
- [ea01] E. Tonti et al. Algebraic Formulation of Physical Fields. <http://discretphysics.dic.units.it>, 2001. home page.

- [Eng90] Erwin Engeler. Combinatory differential fields. *Theoretical Computer Science*, 72(2-3) :119–131, 1990.
- [FB93] Walter Fontana and Leo W. Buss. What would be conserved if "the tape were played twice" ?, April 21 1993.
- [FB94] Walter Fontana and Leo W. Buss. "the arrival of the fittest" : Toward a theory of biological organization. *Bulletin of Mathematical Biology*, 1994.
- [FB96] W. Fontana and L. Buss. *Boundaries and Barriers*, Casti, J. and Karlqvist, A. eds., chapter The barrier of objects : from dynamical systems to bounded organizations, pages 56–116. Addison-Wesley, 1996.
- [Fon92] Walter Fontana. Algorithmic chemistry. In Christopher G. Langton, Charles Taylor, J. Dooyne Farmer, and Steen Rasmussen, editors, *Proceedings of the Workshop on Artificial Life (ALIFE '90)*, volume 5 of *Santa Fe Institute Studies in the Sciences of Complexity*, pages 159–210, Redwood City, CA, USA, February 1992. Addison-Wesley.
- [Fre03] Edward Fredkin. An introduction to digital philosophy. *International Journal of Theoretical Physics*, 42(3) :189–247, 2003.
- [GGMP02] Jean-Louis Giavitto, Christophe Godin, Olivier Michel, and Przemyslaw Prusinkiewicz. *Modelling and Simulation of biological processes in the context of genomics*, chapter "Computational Models for Integrative and Developmental Biology". Hermes, July 2002. <http://www.ibisc.fr/~michel/PUBLIS/2002/autran02.ps.gz>.
- [Gia03] Jean-Louis Giavitto. Invited talk : Topological collections, transformations and their application to the modeling and the simulation of dynamical systems. In *Rewriting Technics and Applications (RTA '03)*, volume LNCS 2706 of *LNCS*, pages 208 – 233, Valencia, June 2003. Springer.
- [Gil77] Daniel T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.*, 81(25) :2340–2361, 1977.
- [GLRT04] Frédéric Gruau, Yves Lhuillier, Philippe Reitz, and Olivier Temam. Blob computing. In *Proceedings of the first conference on computing frontiers on Computing frontiers*, pages 125–139. ACM Press, 2004.
- [GM01a] Jean-Louis Giavitto and Olivier Michel. Declarative definition of group indexed data structures and approximation of their domains. In *PPDP '01 : Proceedings of the 3rd ACM SIGPLAN international conference on Principles and practice of declarative programming*, pages 150–161, New York, NY, USA, 2001. ACM Press. <http://www.ibisc.fr/~michel/PUBLIS/2001/ppdp01.pdf>.
- [GM01b] Jean-Louis Giavitto and Olivier Michel. MGS : Implementing a unified view on four biologically inspired computational models. In *Pre-proceedings of WMC-CdeA 2001 (Workshop on Membrane Computing, Curtea de Arges)*. Research Report 17/01 of the Universitat Rivira I Virgili, Tarragona, Spain, August 2001. <http://www.ibisc.fr/~michel/PUBLIS/2001/wmc01.ps.gz>.
- [GM02a] Jean-Louis Giavitto and Olivier Michel. Data structure as topological spaces. In *Proceedings of the 3rd International Conference on Unconventional Models*

- of Computation UMC02*, volume 2509, pages 137–150, Himeji, Japan, October 2002. <http://www.ibisc.fr/~michel/PUBLIS/2002/umc02.pdf>.
- [GM02b] Jean-Louis Giavitto and Olivier Michel. Pattern-matching and rewriting rules for group indexed data structures. *ACM SIGPLAN Notices*, 37(12) :76–87, December 2002. <http://www.ibisc.fr/~michel/PUBLIS/2003/sigplan03.pdf>.
- [GM02c] Frédéric Gruau and Philippe Malbos. The blob : A basic topological concept for "hardware-free" distributed computation data structure as topological spaces. In *Proceedings of the 3rd International Conference on Unconventional Models of Computation UMC02*, volume 2509, pages 151–163, Himeji, Japan, October 2002. Lecture Notes in Computer Science.
- [GM03] J.-L. Giavitto and O. Michel. Modeling the topological organization of cellular processes. *BioSystems*, 70(2) :149–163, 2003.
- [GMC02] Jean-Louis Giavitto, Olivier Michel, and Julien Cohen. Pattern-matching and rewriting rules for group indexed data structures. In *ACM Sigplan Workshop RULE'02*, pages 55–66, Pittsburgh, October 2002. ACM. <http://www.ibisc.fr/~michel/PUBLIS/2002/rule02.pdf>.
- [GMD01] Jean-Louis Giavitto, Olivier Michel, and Franck Delaplace. Declarative simulation of dynamical systems : the 81/2 programming language and its application to the simulation of genetic networks. In *Proceedings of IPCAT 2001 (Workshop on Information Processing in Cells and Tissues)*, August 2001. <http://www.ibisc.fr/~michel/PUBLIS/2001/ipcat.ps.gz>.
- [GMM04] Jean-Louis Giavitto, Grant Malcolm, and Olivier Michel. Rewriting systems and the modelling of biological systems. *Comparative and Functional Genomics*, 5 :95–99, February 2004. <http://www.ibisc.fr/~michel/PUBLIS/2004/CFG04.pdf>.
- [Gru04] F. Gruau. The "Blob Computing" Home page, 2004. <http://blob.lri.fr>.
- [GS06] Jean-Louis Giavitto and Antoine Spicher. *Systems Self-Assembly : multidisciplinary snapshots*, chapter Simulation of self-assembly processes using abstract reduction systems. Elsevier, 2006.
- [GS08] Jean-Louis Giavitto and Antoine Spicher. Topological rewriting and the geometrization of programming. *Physica D*, 2008. (Article in press, publication scheduled for autumn 2008).
- [GSM07] Jean-Louis Giavitto, Antoine Spicher, and Olivier Michel. Topological rewriting and the geometrization of programming. Technical Report IBISC-XX-2007, IBISC, September 2007.
- [Hor01] Paul Horn. Autonomic computing : IBM's perspective on the state of information technology. Technical report, IBM Research, October 2001. http://www.research.ibm.com/autonomic/manifesto/autonomic_computing.pdf.
- [Mac02a] Bruce MacLennan. Universally programmable intelligent matter : A systematic approach to nanotechnology. In Clifford Lau, editor, *IEEE-Nano 2002*. IEEE, 2002.
- [Mac02b] Bruce J. Maclennan. Replication, sharing, deletion, lists, and numerals : Progress on universally programmable intelligent matter, November 23 2002.

- [Mac03] MIT Project Mac. The Amorphous Computing Home page, 2003. <http://www.swiss.ai.mit.edu/projects/amorphous>.
- [Man04] Nicolas Mann. Hyperstructures et modélisation de chimie artificielle dans le langage MGS. Master's thesis, DEA AMIB Université d'Évry, 2004.
- [Mat00] Claudio Mattiussi. The finite volume, finite element, and finite difference methods as numerical methods for physical field problems. *Advances in Imaging and Electron Physics*, 113 :1–146, 2000. *NOTES* : Very highly recommended. Tonti diagrams. Fits many mimetic approaches into a general framework.
- [MBFG04] O. Michel, J.-P. Banatre, P. Fradet, and J.-L. Giavitto. The Unconventional Programming Paradigms home page (UPP04). <http://upp.lami.univ-evry.fr>, 2004. International workshop for "Challenges, Visions and Research Issues for New Programming Paradigms", 15 - 17 September 2004, Mont Saint-Michel, France.
- [Mic96] Olivier Michel. Design and implementation of 8_{1/2}, a declarative data-parallel language. *Computer Languages*, 22(2/3) :165–179, 1996. <http://www.ibisc.fr/~michel/PUBLIS/1996/ComputerLanguages.pdf>.
- [NU03] NSF and San Diego State University. Mimetic Discretization of Continuum Mechanics. <http://www.sci.sdsu.edu/compscims/MIMETIC>, 2003. First interdisciplinary workshop on mimetic discretizations method.
- [Pau00] Gheorghe Paun. Computing with membranes. *Journal of Computer and System Sciences*, 1(61) :108–143, 2000.
- [PH03] M. Parashar and S. Hariri, editors. *Autonomic Applications Workshop*, Taj Krishna, Hyderabad, India, December 2003. special issue of "Cluster Computing, The Journal of Networks, Software Tools and Applications" on Autonomic Applications.
- [PL90] P. Prusinkiewicz and A. Lindenmayer. *The algorithmic beauty of plants*. Springer-Verlag, New York, 1990. With J. S. Hanan, F. D. Fracchia, D. R. Fowler, M. J. M. de Boer, and L. Mercer.
- [PS93] Richard S. Palmer and Vadim Shapiro. Chain models of physical behavior for engineering analysis and design. *Research in Engineering Design*, 5 :161–184, 1993. Springer International.
- [Req98] Manfred Requardt. Cellular networks as models for planck-scale physics. *J. Phys.*, A31 :7997–8021, 1998.
- [RLS⁺04] John H. Reif, Thomas LaBean, Sudheer Sahu, Hao Yan, and Peng Yin. Design, simulation, and experimental demonstration of self-assembled dna nanostructures and dna motors. In *Computational Modeling and Simulation of Materials (CIMTEC) Conference*, Acireale, Sicily, Italy, May 29-June 4 2004.
- [Roz87] Marie-Noëlle Rozin. Etude pratique des ressources de la programmation plane, July 1987. Mémoire de maitrise sous la direction de P. Greussay.
- [Roz90] Marie-Noëlle Rozin. Opérateurs planaires appliqués à la construction de configurations combinatoires sur processeur simd. 9(5), May 1990.

- [RS92] Grzegorz Rozenberg and Arto Salomaa. *Lindenmayer Systems*. Springer, Berlin, 1992.
- [SM04a] Antoine Spicher and Olivier Michel. Integration and pattern-matching of topological structures in a functional language. In *International Workshop on Implementation and Application of Functional Languages (IFL04)*, Lübeck, September 2004. <http://www.ibisc.fr/~michel/PUBLIS/2004/IFL04.pdf>.
- [SM04b] Antoine Spicher and Olivier Michel. Manipulations de structures topologiques dans un langage déclaratif pour la simulation. In *11ème Journées du GT "Animation et Simulation" (GTAS'2004)*, Reims, juin 2004. AFIG et LERI, Université de Reims. <http://www.ibisc.fr/~michel/PUBLIS/2004/gtas04.pdf>.
- [SM05] Antoine Spicher and Olivier Michel. Declarative modeling of a neurulation-like process. In *Sixth International Workshop on Information Processing in Cells and Tissues*, 2005. http://www.ibisc.fr/~michel/PUBLIS/2005/ipcat05_spicher_michel_CRV.pdf.
- [SM06] Antoine Spicher and Olivier Michel. Declarative modeling of a neurulation-like process. *BioSystems*, 87 :281–288, February 2006. <http://www.ibisc.fr/~michel/PUBLIS/2006/biosystem06.pdf>.
- [SM07] Antoine Spicher and Olivier Michel. Représentation et manipulation de structures topologiques dans un langage fonctionnel. *Technique et Science Informatique*, 2007. <http://www.ibisc.fr/~michel/PUBLIS/2007/tsijfla07.pdf>.
- [SMC⁺08] Antoine Spicher, Olivier Michel, Mikolaj Cieslak, Jean-Louis Giavitto, and Przemyslaw Prusinkiewicz. Stochastic p systems and the simulation of biochemical processes with dynamic compartments. *BioSystems*, 91(3) :458–472, March 2008.
- [SMG04a] Antoine Spicher, Olivier Michel, and Jean-Louis Giavitto. A topological framework for the specification and the simulation of discrete dynamical systems. Technical Report LaMI-99-2004, LaMI, May 2004. <http://www.ibisc.fr/~michel/PUBLIS/2004/rt-99-2004.pdf>.
- [SMG04b] Antoine Spicher, Olivier Michel, and Jean-Louis Giavitto. A topological framework for the specification and the simulation of discrete dynamical systems. In *Sixth International conference on Cellular Automata for Research and Industry (ACRI'04)*, volume 3305 of *LNCS*, Amsterdam, October 2004. Springer. <http://www.ibisc.fr/~michel/PUBLIS/2004/acri04.pdf>.
- [SMG05] Antoine Spicher, Olivier Michel, and Jean-Louis Giavitto. Algorithmic self-assembly by accretion and by carving in MGS. In *7th International Conference on Artificial Evolution*, 2005. <http://www.ibisc.fr/~michel/PUBLIS/2005/ea05.pdf>.
- [Sor91] Rafael D. Sorkin. A finitary substitute for continuous topology. *Int. J. Theor. Phys.*, 30 :923–948, 1991.
- [Spi03] Antoine Spicher. Typage et compilation de filtrage de chemins dans des collections topologiques. Master's thesis, DEA AMIB Université d'Évry, 2003.
- [Spi06] Antoine Spicher. *Transformation de collections topologiques de dimension arbitraire – Application à la modélisation de systèmes dynamiques*. PhD thesis, Université d'Évry, December 2006.

- [TM87] T. Toffoli and N. Margolus. *Cellular automata machines : a new environment for modeling*. MIT Press, Cambridge, 1987.
- [Ton74] Enzo Tonti. The algebraic-topological structure of physical theories. In P. G. Glockner and M. C. sing, editors, *Symmetry, similarity and group theoretic methods in mechanics*, pages 441–467, Calgary, Canada, August 1974.
- [VMU74] F. J. Varela, H. R. Maturana, and R. Uribe. Autopoiesis : The organization of living systems. *BioSystems*, 5(4) :187–196, 1974.
One of the first presentations of the concept of autopoiesis.
- [WER01] Erik Winfree, Tony Eng, and Grzegorz Rozenberg. String tile models for dna computing by self-assembly. In *Revised Papers from the 6th International Workshop on DNA-Based Computers*, pages 63–88. Springer-Verlag, 2001.