

# CURRICULUM VITAE

Nom : **GOURAUD** Sandrine-Dominique

Mots-clefs : *test (statistique, structurel ou fonctionnel) de logiciels,  
génération uniforme de structures combinatoires,  
résolution de contraintes,  
systèmes de réécriture*



# Table des matières

<b>CURRICULUM VITAE</b>	<b>7</b>
<b>1 État civil</b>	<b>7</b>
1.1 Situation professionnelle . . . . .	7
1.2 Situations antérieures . . . . .	7
1.3 Sujets de recherche . . . . .	7
<b>2 Formation</b>	<b>8</b>
<b>PUBLICATIONS</b>	<b>9</b>
<b>3 Articles publiés</b>	<b>9</b>
3.1 Conférences internationales de large diffusion avec comité de lecture sur texte complet . . . . .	9
3.2 Conférences nationales avec comité de lecture sur texte complet . . . . .	9
3.3 Mémoires . . . . .	10
<b>4 Article en cours de soumission</b>	<b>10</b>
<b>5 Article en cours de rédaction</b>	<b>10</b>
<b>ENSEIGNEMENT</b>	<b>11</b>
<b>6 Activités d'enseignement</b>	<b>11</b>
6.1 Description des filières spécialisées . . . . .	11
6.1.1 IUP/CFA MIAGE . . . . .	11
6.1.2 FIIFO . . . . .	12
6.2 Description des enseignements . . . . .	12
6.2.1 Module <i>Principes d'Interprétation des Langages</i> . . . . .	12
6.2.2 Module <i>Introduction au Génie Logiciel</i> . . . . .	12
6.2.3 Module <i>Pratique d'Unix</i> . . . . .	13
6.2.4 Module <i>Combinatoire pour l'informatique</i> . . . . .	13
6.2.5 Module <i>Analyse et Conception des Systèmes d'Information</i> . . . . .	13
6.2.6 Module <i>Génie Logiciel</i> . . . . .	14
6.2.7 Module <i>Algorithmique</i> . . . . .	14
6.2.8 Module <i>Approche Impérative</i> . . . . .	14
6.3 Encadrements divers . . . . .	14
6.3.1 Encadrement de stage . . . . .	14
6.3.2 Tutorat . . . . .	15

<b>7</b>	<b>Projet d'enseignement</b>	<b>15</b>
	<b>VULGARISATION ET CHARGES</b>	<b>17</b>
<b>8</b>	<b>Vulgarisation scientifique</b>	<b>17</b>
8.1	Projet Scientifique Parrainé (2001 à 2003) . . . . .	17
8.1.1	Cadre . . . . .	17
8.1.2	Description . . . . .	17
8.2	La fête de la Science (2000 et 2002) . . . . .	18
8.2.1	Cadre . . . . .	18
8.2.2	Description . . . . .	18
<b>9</b>	<b>Charges collectives</b>	<b>19</b>
9.1	Relecture d'articles . . . . .	19
9.2	Participation aux portes ouvertes . . . . .	19
9.3	Gestion de pages Web . . . . .	19
9.4	Comité d'organisation . . . . .	19
	<b>RECHERCHE</b>	<b>21</b>
<b>10</b>	<b>Activités de recherche</b>	<b>21</b>
10.1	Recherche post-doctorale (2004-2005) . . . . .	21
10.2	Recherche doctorale (2000 à 2004) . . . . .	22
10.2.1	Le test statistique . . . . .	22
10.2.2	Génération aléatoire de structures combinatoires . . . . .	23
10.2.3	Utilisation des structures combinatoires pour le test statistique . . .	23
10.2.4	Conclusion . . . . .	24
<b>11</b>	<b>Projet de recherche</b>	<b>24</b>
11.1	Perspectives de recherche doctorale . . . . .	24
11.1.1	Mieux exploiter les techniques combinatoires . . . . .	25
11.1.2	Limiter l'influence des chemins infaisables . . . . .	25
11.1.3	Autres . . . . .	26
11.2	Perspectives de recherche post-doctorale . . . . .	26
11.2.1	Utilisation des structures combinatoires . . . . .	26
11.2.2	Extension aux langages fonctionnels . . . . .	27
11.3	Perspectives générales . . . . .	27
<b>12</b>	<b>Participation à des groupes de travail</b>	<b>27</b>
12.1	Projet RNTL CASTLES (2004) . . . . .	27
12.2	Projet national sur le Test de Robustesse (2001-2002) . . . . .	27

<b>13 Exposés et participation à des colloques</b>	<b>28</b>
13.1 Exposés à des conférences . . . . .	28
13.2 Autres participations à des conférences . . . . .	28
<b>14 Séminaires</b>	<b>28</b>
<b>15 Développement</b>	<b>28</b>
15.1 JSL2CHR . . . . .	28
15.2 AuGuSTe ( <i>Automated Generation of Statistical Testing</i> ) . . . . .	29
15.3 SiRéMa ( <i>Simulateur de Réseau Manhattan</i> ) . . . . .	29
<b>AUTRES</b>	<b>31</b>
<b>16 Formations complémentaires</b>	<b>31</b>
16.1 École Doctorale . . . . .	31
16.2 Centre d'Initiation à l'Enseignement Supérieur (CIES) . . . . .	31
<b>17 Expériences professionnelles</b>	<b>32</b>
<b>18 Activités culturelles et sportives</b>	<b>32</b>



# CURRICULUM VITAE

## 1 État civil

GOURAUD Sandrine-Dominique  
27 ans  
Née le 11 avril 1977 à Paris XIV  
Mariée  
Nationalité Française

### 1.1 Situation professionnelle

J'effectue de la recherche post-doctorale à l'IRISA (Rennes) dans le projet LANDE. Mon contrat s'arrête au 31 août 2005.

CNRS-IRISA  
Campus Universitaire de Beaulieu  
35042 RENNES Cedex  
Tél.(bureau) : 02.99.84.75.86  
Mél. : gouraud@irisa.fr  
<http://www.irisa.fr/lande/gouraud/>

### 1.2 Situations antérieures

**2003-2004** Demi-Attachée Temporaire d'Enseignement et de Recherche (ATER),  
Université Paris XI

**2000 à 2003** Allocataire Monitrice, Université Paris XI

### 1.3 Sujets de recherche

Dans la nomenclature thématique proposée par la CNU 27, mes sujets de recherche sont :

- Génie logiciel et programmation,
- Approches formelles (spécification, validation, vérification),
- Programmation déclarative (fonctionnelle, logique).

Plus précisément, je travaille sur les sujets suivants :

- Test de logiciels : statistique, structurel ou fonctionnel,
- Génération uniforme de structures combinatoires,
- Résolution de contraintes,
- Systèmes de réécriture.

## 2 Formation

**2004 Doctorat** d'Informatique

Thèse : *Utilisation des Structures Combinatoires pour le Test Statistique*

Thèse de l'université Paris XI, soutenue le 24 juin 2004 à Orsay.

Mention : **Très Honorable**.

Directrice de thèse : Marie-Claude Gaudel

Composition du jury :

A. Denise (LRI, Université Paris XI),

P. Flajolet (rapporteur, INRIA Rocquencourt),

B. Marre (CEA Saclay),

F. Ouabdesselam (rapporteur, LSR-IMAG, Grenoble),

P. Thévenod-Fosse (présidente, LAAS Toulouse).

**2000 DEA** Programmation : Sémantique, Preuves et Langages

Mention **Assez-Bien**, Université Paris XI

Responsable : Marie-Claude Gaudel

Sujet : *Application des Techniques de Génération Aléatoire au Test de Logiciel*

**1999 Maîtrise** d'Informatique, Mention **Bien**, Université Paris XI

**1998 Licence** d'Informatique, Mention **Bien**, Université Paris XI

**1997 DEUG** MIAS, Mention Assez-Bien, Université Paris XI

**1995 Baccalauréat** Série Scientifique, Mention Passable, Lycée H. Boucher, Paris

# PUBLICATIONS

Toutes ces publications sont disponibles sur ma page Web :  
<http://www.irisa.fr/lande/gouraud/>

## 3 Articles publiés

### 3.1 Conférences internationales de large diffusion avec comité de lecture sur texte complet

- [1] A. Denise, M.-C. Gaudel et S.-D. Gouraud,  
A Generic Method for Statistical Method,  
In *Fifteenth IEEE International Symposium on Software Reliability Engineering (ISSRE)*,  
pages 25-34, 2004, Saint Malo  
Taux de sélection : 39/120
- [2] S.-D. Gouraud, A. Denise, M.-C. Gaudel et B. Marre,  
A New Way of Automating Statistical Testing Methods,  
In *Sixteenth IEEE International Conference on Automated Software Engineering (ASE)*, IEEE Computer Society Press, pages 5-12, 2001, San Diego (USA)  
Taux de sélection : 32/164

### 3.2 Conférences nationales avec comité de lecture sur texte complet

- [3] S.-D. Gouraud et A. Gotlieb,  
Using CHRs to generate functional test cases for the Java Card Virtual Machine,  
*Journées Francophones de Programmation par Contraintes (JFPC) (à paraître)*
- [4] S.-D. Gouraud,  
AuGuSTe : un outil pour le Test Statistique,  
*Approches Formelles dans l'Assistance au Développement de Logiciels (AFADL)*,  
pages 337-340, 2004, Besançon
- [5] S.-D. Gouraud,  
Application de la Génération Aléatoire de Structures Combinatoires au Test de Logiciel,  
*AFADL*, pages 99-112, 2001, Nancy

### 3.3 Mémoires

- [6] S.-D. Gouraud,  
Utilisation des Structures Combinatoires pour le Test Statistique,  
mémoire de doctorat, Université Paris XI
- [7] S.-D. Gouraud,  
Application des Techniques de Génération Aléatoire au Test de Logiciel,  
mémoire de DEA, Université Paris XI

## 4 Article en cours de soumission

- [8] S.-D. Gouraud,  
AuGuSTe : a Tool for Statistical Testing (Experimental Results),  
*Third IEEE International Conference on Software Engineering and Formal Methods (SEFM)*,  
RR-LRI-1400

## 5 Article en cours de rédaction

- [9] N. Baskiotis, M. Sebag, M.-C. Gaudel et S.-D. Gouraud  
Learning Enhanced Software Testing  
Cet article présente une première utilisation des techniques d'inférence et d'apprentissage pour le test (cf. p.26).

# ENSEIGNEMENT

## 6 Activités d'enseignement

En tant que chargée de Travaux Dirigés (TD) et de Travaux Pratiques (TP), j'ai participé à la rédaction de sujets d'examens/partiels, à la surveillance d'examens/partiels, à la correction de copies de partiels/TP, à la mise à jour d'énoncés de TD/TP, à la rédaction d'un projet de programmation et à différents jurys. Tous les horaires suivants sont donnés en heures équivalent TD.

Année	Filière	Niveau	Intitulé du cours	TD	TP
2003/2004	IUP MIAGE	Bac+2	Principes d'interprétation des langages	26	10
	IUP MIAGE	Bac+3	Introduction au génie logiciel	33	
	Licence	Bac+3	Pratique d'unix		21
	CFA MIAGE	Bac+4	Encadrement d'une apprentie	6	
2002/2003	DEUG	Bac+2	Principes d'interprétation des langages		10
	Licence	Bac+3	Introduction au génie logiciel	33	
2001/2002	DEUG	Bac+2	Combinatoire pour l'informatique	25	21
	FIIFO	Bac+2	ACSI	12	
	FIIFO	Bac+2	Encadrement d'un étudiant	2	
	FIIFO	Bac+5	Génie logiciel	8	
2000/2001	DEUG	Bac+2	Principes d'interprétation des langages	26	
	DEUG	Bac+2	Approche impérative		36
	CFA MIAGE	Bac+3	Algorithmique		7
	CFA MIAGE	Bac+3	Encadrement de quatre apprentis	4	
	Maîtrise	Bac+4	Encadrement d'un TER stage	3	
TOTAL en heures eqTD				283	

### 6.1 Description des filières spécialisées

#### 6.1.1 IUP/CFA MIAGE

L'IUP MIAGE (Méthodes Informatiques Appliquées à la Gestion des Entreprises) est une formation professionnelle en trois ans qui a pour objectif de former des cadres en informatique de gestion de niveau bac+4. À la fin de leurs études, les ingénieurs-maîtres MIAGistes sont capables de concevoir et de mettre en oeuvre des systèmes d'information, de conduire des projets et d'encadrer des équipes. Une description plus détaillée est disponible sur le site Web <http://www.isi.u-psud.fr/>

Le CFA (Centre de Formation d'Apprentis) MIAGE est la formation par apprentissage des ingénieurs-maîtres MIAGistes.

### 6.1.2 FIIFO

La FIIFO (Formation d'Ingénieurs en Informatique de la Faculté d'Orsay) est une formation en cinq ans qui a pour l'objectif de former des ingénieurs informaticiens. La mission de ces nouveaux ingénieurs consiste en la mise en oeuvre des techniques informatiques au sein d'une entreprise (conception, développement, mise au point des logiciels; conception de la documentation; validation des performances d'exploitabilité et prise en charge des évolutions techniques et fonctionnelles). Une description plus détaillée est disponible sur le site Web <http://www.fiifo.u-psud.fr/>

## 6.2 Description des enseignements

### 6.2.1 Module *Principes d'Interprétation des Langages*

Ce module fut dirigé par Claude Marché en 2000/2001 puis par Brigitte Rozoy en 2002/2003 et 2003/2004. Il consiste à introduire les notions d'expressions régulières et de grammaires ainsi que des techniques de contrôle de types et d'évaluation des langages.

J'ai enseigné ce module (TD comme TP) aux étudiants d'IUP MIAGE et de DEUG MIAS. Dans ce cadre, j'ai participé :

- à la rédaction (création ou mise à jour) des feuilles de TD, des feuilles de TP et des sujets de partiels et d'examens,
- à la correction des copies de partiels et des comptes-rendus de TP,
- à la surveillance des partiels et des examens.

J'aime enseigné ce module car il permet de faire découvrir aux étudiants les bases des langages de programmation. C'est aussi le module qui est à l'origine de mon orientation professionnelle vers l'informatique d'où mon affinité particulière à l'enseigner.

### 6.2.2 Module *Introduction au Génie Logiciel*

Ce module dirigé par Marie-Claude Gaudel consiste à initier les étudiants de la licence d'informatique et de l'IUP MIAGE au génie logiciel. En particulier, nous y étudions des méthodes orientées objet d'analyse et de conception où la méthode UML et le langage Java servent de support. Puis, nous terminons par une présentation des techniques structurales de test et par une initiation à la preuve de programme (logique de Hoare).

J'ai enseigné ce module aux étudiants d'IUP et de licence d'informatique. Dans ce cadre, j'ai participé :

- à la relecture des feuilles de TD et des sujets de partiels et d'examens
- à la correction des copies de partiels
- à la surveillance des partiels et examens
- à la gestion de la page Web du module

C'est un module difficile à enseigner car les étudiants ne voient pas toujours l'intérêt de ces techniques (en particulier à court terme). De plus, ils sont souvent gênés par l'absence d'une solution unique et donc d'une solution identique pour tous les groupes de TD. C'est un module qui demande beaucoup de travail personnel.

### 6.2.3 Module *Pratique d'Unix*

Ce module dirigé par Nicolas Bredèche consiste à introduire les bases d'Unix à partir d'un cours magistral et de TP aux étudiants de licence d'informatique et d'IUP MIAGE. Nous y étudions en particulier l'organisation des fichiers, les droits d'accès, les processus, les outils de développement logiciel, les éditeurs et la gestion de versions.

J'ai enseigné ce module aux étudiants d'IUP et de licence d'informatique. Dans ce cadre, j'ai participé :

- à la relecture des TP
- à la rédaction et surveillance des TP notés (partiel et examen)
- à la correction des copies des TP notés (partiel et examen)

C'est un module très intéressant qui permet aux étudiants (et aux enseignants) de découvrir puis de se débrouiller dans un environnement Unix/Linux. Malheureusement, je n'ai pas pu profiter pleinement de cet enseignement. En effet, j'ai souvent remplacé mes collègues et n'ai donc effectué que les trois premiers TP avec trois groupes de TP différents.

### 6.2.4 Module *Combinatoire pour l'informatique*

Ce module dirigé par Laurent Rosaz consiste à introduire les notions de démonstration et de récurrence ainsi que la logique propositionnelle et la logique des prédicats.

J'ai enseigné ce module aux étudiants de DEUG MIAS. Dans ce cadre, j'ai participé :

- à la relecture des feuilles de TD et des sujets de partiels et d'examens
- à la correction des copies de partiels
- à la surveillance des partiels et examens

### 6.2.5 Module *Analyse et Conception des Systèmes d'Information*

Ce module dirigé par Marie-Claude Heydemann consiste à introduire les notions d'analyse et de conception des systèmes d'information aux étudiants de FIIFO 2ième année. Nous étudions les différents niveaux d'analyse, la modélisation des données (construction du dictionnaire des données, dépendances fonctionnelles, construction et validation d'un schéma entité-association (E/A), passage du modèle E/A au modèle relationnel), la modélisation des traitements et les outils de la méthode MERISE.

J'ai enseigné ce module aux étudiants de FIIFO. Dans ce cadre, j'ai participé :

- à la relecture des feuilles de TD, de la feuille de projet et des sujets de partiels et d'examens
- à la correction des copies de partiels et de projet
- à la surveillance du partiel
- au jury

Pour les mêmes raisons que le module *Introduction au Génie Logiciel*, c'est un module difficile à enseigner.

### **6.2.6 Module *Génie Logiciel***

Ce module dirigé par Frédéric Voisin consiste à introduire en 8h de TD les notions de tests et de preuves de programmes à des étudiants de dernière année de FIIFO.

Ce public est constitué à la fois d'étudiants issus de la formation principale et d'ingénieurs déjà en poste qui élargissent leurs compétences. C'est la première fois que j'ai dû enseigner à des gens plus âgés que moi. Bien que courte, c'est une expérience qui s'est avérée très enrichissante humainement.

### **6.2.7 Module *Algorithmique***

Ce module dirigé par Jean-Pierre Tillich consiste à introduire l'algorithmique à des étudiants de CFA MIAGE. Nous avons étudié les techniques fondamentales de l'algorithmique, les modes de tri, les structures arborescentes, la recherche d'information, les algorithmes de parcours de graphes et la pratique d'un langage informatique orienté objet. Dans ce module, j'ai assisté les étudiants dans la conception de leur projet puis corrigé ces projets.

### **6.2.8 Module *Approche Impérative***

Ce module dirigé par Michel Beaudouin-Lafon consiste à introduire la programmation impérative aux étudiants de DEUG. Nous avons étudié un langage de réalisation, les structures de données, la syntaxe et la sémantique (mémoire, environnement, passage de paramètre, pointeurs).

J'ai enseigné ce module aux étudiants de DEUG MIAS. Dans ce cadre, j'ai participé :

- à la relecture des feuilles de TD
- à l'élaboration d'une feuille de TP
- à l'élaboration de la feuille de projet
- à la correction du projet
- aux soutenances des projets

Ce module m'a offert une expérience particulière dans la gestion des ressources humaines. En effet, comme l'équipe pédagogique était composée de douze personnes, il fallait être organisé (aucun retard possible) et souple afin que les décisions soient rapidement unanimement prises.

## **6.3 Encadrements divers**

### **6.3.1 Encadrement de stage**

**2003-2004** Tutrice d'une apprentie en CFA MIAGE dernière année.

Dans ce cadre, j'ai :

- pris contact régulièrement avec mon étudiante pour m'assurer du bon déroulement de son stage et de ses études,
- rempli régulièrement le livret d'apprentissage,
- visité mon étudiante dans son entreprise,

- rencontré son maître de stage,
- relu son mémoire de stage.

**2001-2002** Tutrice d'un étudiant de FIIFO 2ième année.

Dans ce cadre, j'ai :

- pris contact régulièrement avec mon étudiant pour m'assurer du bon déroulement de son stage,
- visité mon étudiant dans son entreprise,
- rencontré son maître de stage,
- relu son mémoire de stage,
- assisté à sa soutenance.

**2000-2001** Co-encadrante d'un binôme TER Stage sur des travaux faisant intervenir les logiciels SiRéMa (cf. p.29) et Grafomatik.

Les TER (Travaux d'Étude et de Recherche) sont de gros projets effectués par au moins deux étudiants pendant tout un semestre.

Dans ce cadre, j'ai :

- encadré ces étudiants,
- relu leur mémoire de stage.

### 6.3.2 Tutorat

**2000-2001** Tutrice de quatre apprentis en CFA MIAGE.

Mon travail de tuteur consistait à prendre régulièrement contact avec mes étudiants pour m'assurer du bon déroulement de leur stage.

**Rentrées 1999 et 2000** Tutorat d'accueil pour les nouveaux étudiants de DEUG, Université Paris XI.

Organisé par le SCUIO (Service Commun Universitaire d'Information et d'Orientation), l'objectif de ce tutorat est de présenter la faculté d'Orsay (bibliothèque, lieux de restauration, bâtiments administratifs importants, etc.) à tous les nouveaux étudiants et de répondre à leurs premières questions.

**Semestre 1 1998-1999** Tutorat d'informatique pour les étudiants de DEUG, Université Paris XI.

Dirigé par Philippe Chatalic, l'objectif de ce tutorat était d'aider les étudiants de DEUG MIAS dans la discipline informatique.

## 7 Projet d'enseignement

Au cours de ma thèse, j'ai eu l'opportunité d'enseigner

- dans différentes filières (ingénieur FIIFO, IUP MIAGE, DEUG, licence),

- dans différents types de formation (initiale ou par apprentissage)
- et à différents niveaux (de bac+2 à bac+5).

Ce large spectre m'a permis de découvrir l'hétérogénéité de la population estudiantine et me permet maintenant d'enseigner sans appréhension à n'importe quel public.

L'enseignement que j'ai effectué dans diverses disciplines et les connaissances que j'ai acquises au cours de mes recherches, font que je me sens capable d'effectuer des enseignements diversifiés tels que

- l'algorithmique,
- la compilation,
- les langages fonctionnels (OCaml),
- les langages impératifs (Pascal, C),
- les langages orientés objet (Java),
- la programmation logique (Prolog),
- la résolution de contraintes,
- le test,
- la preuve,
- l'analyse de systèmes d'information avec l'utilisation de méthodes comme UML ou Merise.

Ma formation généraliste en Informatique me permet également d'enseigner dans d'autres domaines.

# VULGARISATION ET CHARGES

## 8 Vulgarisation scientifique

### 8.1 Projet Scientifique Parrainé (2001 à 2003)

#### 8.1.1 Cadre

Les Projets Scientifiques Parrainés (PSP) ont été créés en 2000 par l'Inspection Académique de l'Essonne. Ces projets reposent sur le développement d'un partenariat entre les grands pôles scientifiques et les établissements scolaires (primaires, collèges, lycées) autour de plusieurs objectifs :

- la sensibilisation des élèves à une démarche scientifique devant les faire évoluer dans leur processus d'apprentissage et d'acquisition,
- la mise en place efficace de projets pédagogiques permettant aux élèves de contextualiser les apprentissages,
- le développement de la motivation des élèves, de l'envie d'apprendre par la réalisation d'un projet concret et attrayant,
- l'évolution des pratiques pédagogiques.

Plus de détails peuvent être trouvés sur le site Web :

<http://www.ac-versailles.fr/ia91/pedagogie/psp-principhisto.htm>

Dans ce cadre, Alexandre Termier et moi avons commencé un projet d'éveil à l'informatique avec Anne-Marie Rullier, enseignante d'une classe de CP-CE1 à l'École des Verts Prés, rue Verts Prés, 91620 Nozay.

#### 8.1.2 Description

Au cours de ces deux ans, nous avons abordé

- l'architecture de l'ordinateur : nous avons ouvert un ordinateur et les élèves ont recherché les éléments qu'ils connaissaient comme le processeur, le disque dur, la carte mère, la RAM, la carte vidéo ;
- les différents supports de stockage en présentant aux élèves les ancêtres de la disquette et du CD-ROM : nous avons apporté en particulier différents types de bandes et de disquettes. Nous avons aussi étudié l'intérieur d'une disquette et d'un disque dur ;
- la traduction décimale vers binaire, la traduction binaire vers décimale et l'addition avec retenue de nombres binaires. Les nombres binaires (jusqu'à six bits) ont été abordé à l'aide de lampe électrique (éteinte pour 0, allumée pour 1)
- la simulation du fonctionnement de l'ordinateur : des élèves jouaient le rôle du processeur, de l'écran, du clavier mais aussi de canaux de communication entre les différents

éléments. Cet ordinateur-humain était une calculatrice de nombres décimales puis binaires.

- la programmation : le but était de diriger un robot (simulé par Alexandre) d’un bout à l’autre de la classe en utilisant des instructions basiques telles que `fait_un_pas`, `tourne_à_droite`, `tourne_à_gauche` et `stop`. Les élèves écrivaient le programme et le robot exécutait les instructions (quitte à rentrer dans le mur ou les tables).

Cette expérience nous a appris

1. à adapter notre savoir et notre vocabulaire à notre auditoire (élèves comme institutrice) ;
2. à gérer les groupes (maintient de l’ordre en l’absence de l’institutrice).

Les retombées pédagogiques ont été plus que positives sur notre manière d’enseigner à l’université.

## 8.2 La fête de la Science (2000 et 2002)

### 8.2.1 Cadre

Les manifestations rassemblées sous le terme “La fête de la Science : la Science près de Chez Vous” existent depuis 1991. Leur objectif est de susciter le désir d’apprendre et de comprendre les sciences au plus grand nombre ainsi que de convaincre les jeunes de s’engager dans les études scientifiques. Ces manifestations peuvent être des expositions, des bars des sciences, des spectacles, des conférences, des visites de laboratoire, etc. Plus de détails sur ces manifestations peuvent être trouvés sur le site du ministère de la recherche : <http://www.recherche.gouv.fr/manif/index.htm>

Dans ce cadre, les chercheurs du LRI animent un stand thématique à la ferme du Moulon, 91190 Gif Sur Yvette. J’ai participé à cette manifestation en 2000 et en 2002.

### 8.2.2 Description

En 2000, j’ai animé pendant trois jours le stand intitulé “Simulation et analyse de trafic sur un réseau” qui était une présentation de l’outil SiRéMa (cf. 29). Afin d’expliquer l’outil, j’ai conçu un poster explicatif et ai monté une démonstration.

En 2002, j’ai participé à l’organisation et à animation du stand intitulé “Communication Toute Génération” qui présentait les dernières innovations dans le domaine des Interfaces Hommes-Machines.

Ces deux expériences m’ont permis

- de rencontrer le personnel de la communication de l’université d’Orsay et d’apprendre à concilier leurs intérêts et mes intérêts scientifiques ;
- de rencontrer de manière conviviale des chercheurs d’autres laboratoires du campus ;
- d’expliquer la recherche en informatique au grand public et donc d’adapter mon discours aux différents niveaux scolaires rencontrés.

## 9 Charges collectives

### 9.1 Relecture d'articles

**2005** Relecture de deux articles pour les *Journées Francophones de Programmation par Contraintes (JFPC)*

**2005** Relecture d'un article pour le premier *Workshop on Bytecode Semantics, Verification, Analysis and Transformation (BYTECODE)*

### 9.2 Participation aux portes ouvertes

**1998 à 2004** Portes ouvertes de l'Université Paris XI.

J'ai participé pendant six ans à l'animation du stand du département informatique. L'objectif était de présenter les différentes filières informatiques aux lycéens et aux étudiants de l'université.

### 9.3 Gestion de pages Web

**2000 à 2003** Responsable de la page Web de l'équipe "Programmation et Génie Logiciel" Ma mission était de garder cette page Web à jour : gestion des membres de l'équipe, annonce des séminaires, mise à disposition de liens utiles.

**2002 à 2004** Responsable de la page Web du module "Introduction au Génie Logiciel" J'ai mis à disposition des élèves des énoncés de TD, les corrections ainsi que divers documents qui leur étaient utiles.

### 9.4 Comité d'organisation

**2002** Membre du comité d'organisation de la première journée de l'école doctorale d'informatique d'Orsay. <http://www.lri.fr/~aze/EcoleDoctorale/>

Avec Laura Monceaux (LIMSI), Haifa Zargayouna (LIMSI), Jérôme Azé (LRI), David Mas (LIMSI) et Alexandre Termier, j'ai organisé une rencontre entre les doctorants informaticiens du LRI et du LIMSI. Dans ce cadre, j'ai participé

- à la mise en place du programme de la journée,
- au recrutement des orateurs,
- à la commande du buffet,
- à l'animation d'une table ronde sur les modules de l'école doctorale
- à la mise en place des élections des représentants des doctorants à l'école doctorale.



# RECHERCHE

## 10 Activités de recherche

### 10.1 Recherche post-doctorale (2004-2005)

Ma recherche post-doctorale s'effectue dans le cadre du projet RNTL CASTLES (Conception d'Analyses Statiques et de Tests pour le Logiciel Embarqué Sécurisé) qui réunit des partenaires académiques (IRISA, INRIA Sophia-Antipolis) et des industriels (AQL, Oberthur). L'objectif du projet est de définir un environnement pour l'automatisation de la certification de la plate-forme et des applications Java Card (version réduite de Java avec des extensions propres aux cartes à puce), et d'évaluer cet environnement dans le contexte d'une certification industrielle.

Dans ce projet, notre travail consiste à proposer des techniques (et éventuellement un prototype) de génération automatique de tests de couverture et de conformité de la machine virtuelle Java Card (JCVM) en s'appuyant sur une spécification formelle de chaque instruction de la JCVM donnée en langage Jakarta Specification Language (JSL) et sur l'utilisation des techniques standard de résolution de contraintes.

Le langage JSL est un langage algébrique de premier ordre, avec un système de types polymorphe, basé sur la réécriture conditionnelle. L'originalité de notre approche consiste à utiliser le langage CHR (Constraint Handling Rules) introduit par Frühwirth. Il s'agit d'un langage de règles de réécriture destiné à l'implantation de solveurs de contraintes. Ces règles gardées permettent de réécrire des contraintes pour les simplifier jusqu'à leur résolution. Il existe trois types de règles : la simplification qui permet de remplacer des contraintes tout en préservant l'équivalence logique du système, la propagation qui permet de rajouter des contraintes redondantes pouvant déclencher des simplifications et la simplification qui est une combinaison des deux précédentes.

Une première étape de mes travaux consistait à étudier la conversion d'une spécification formelle JSL en un système de contraintes en s'appuyant sur les CHR, de sorte à obtenir un modèle exécutable de la spécification à partir duquel une génération automatique de test serait possible. Suite à cette étude, j'ai défini un ensemble de règles de traduction permettant cette conversion. Une implémentation est même disponible pour l'environnement Prolog.

Actuellement, j'étudie la prise en compte des objectifs de couverture ou de détection de non-conformités qui peut s'effectuer soit au niveau de la spécification JSL soit par un système de contraintes.

Finalement, l'ensemble des contraintes issues de la spécification JSL et des objectifs de tests constituera la base de notre technique de génération automatique de tests qui devrait s'appuyer sur la résolution de contraintes et la génération aléatoire de données de tests.

Pour ces travaux, les compétences que j’ai acquises au cours de mon DEA (Sémantiques, Preuves et Langages) et de mon doctorat (enseignement et recherche) dans des domaines tels que la spécification formelle, le test, la résolution de contraintes ou les systèmes de réécriture sont particulièrement bien utiles. Ces travaux me permettent d’élargir mes compétences dans le domaine des contraintes et dans celui de l’analyse statique à travers les échanges avec les différents partenaires du projet et les séminaires organisés par mon équipe d’accueil. C’est également une première expérience de travail collaboratif avec des industriels.

## 10.2 Recherche doctorale (2000 à 2004)

Dans ma thèse, je propose une nouvelle approche pour le test statistique de logiciel à partir d’une description sous forme de graphe des comportements du système à tester (graphe de contrôle, statechart, etc.). Son originalité repose sur la combinaison de résultats et d’outils de combinatoire (CS ou MuPAD-combinat) et d’un solveur de contraintes randomisé, pour obtenir une méthode de test complètement automatisée. Contrairement aux approches classiques, la génération aléatoire uniforme est utilisée pour tirer des chemins parmi un ensemble de chemins d’exécution<sup>1</sup> potentiels ou de traces<sup>2</sup> du système à tester. Puis, une étape de résolution de contraintes est utilisée pour déterminer les entrées qui permettront d’exécuter ces chemins.

### 10.2.1 Le test statistique

Le principe du test statistique est de tirer des données dans le domaine d’entrée selon une certaine distribution. Dans sa version classique, la distribution est uniforme mais elle peut également être estimée en fonction de l’exploitation future du système à tester : c’est le test aléatoire opérationnel. L’avantage des méthodes de test statistique est qu’à partir du moment où cette méthode est automatisée, elle permet de faire du test plus intensif que les méthodes déterministes. Le test aléatoire uniforme permet d’assurer une bonne couverture des programmes dont les chemins d’exécution ont des domaines d’entrée de tailles (et donc de probabilités) comparables. Cependant il existe des cas particuliers (comme les cas d’exception ou les valeurs aux bornes) qui correspondent à des sous-domaines d’entrée de probabilités trop faibles et qui sont difficiles à atteindre.

La qualité d’un jeu de test peut s’évaluer à sa capacité à couvrir un critère de couverture. Parmi les critères de couverture les plus connus, on trouve “tous les chemins”, “tous les sommets (resp. arcs)” du graphe représentant le code d’un programme ou sa spécification, etc. Dans le cas d’une méthode de test statistique, cette qualité est liée à la probabilité minimale qu’a un élément du critère d’être atteint.

---

<sup>1</sup>Un chemin d’exécution d’un programme est la suite d’instructions exécutées pour les valeurs d’entrée considérées.

<sup>2</sup>Une trace est une séquence d’entrées/sorties du système. Cela correspond à un chemin de l’automate à entrées et sorties qui modélise une spécification du système à tester.

## 10.2.2 Génération aléatoire de structures combinatoires

L'étude et la simulation des processus stochastiques a largement profité des progrès dans le domaine de la génération aléatoire de structures combinatoires. Une structure combinatoire est soit un objet atomique (de taille 0 ou 1), soit un objet obtenu en combinant à l'aide d'opérateurs (union, produit) des objets atomiques. Les travaux de Wilf et Nijenhuis, puis ceux de Flajolet, Zimmermann et Van Cutsem, constituent la base d'un outil puissant de génération uniforme d'objets complexes comme des graphes, des arbres, des mots, des chemins... Dans ma thèse, j'explore l'usage de tels concepts et outils pour le test de logiciel.

## 10.2.3 Utilisation des structures combinatoires pour le test statistique

Je propose une méthode pour le test statistique basée sur le tirage de chemins dans un graphe en utilisant la génération aléatoire de structures combinatoires. Cette approche a pour objectif de garder les bénéfices du test statistique en y prenant mieux en compte les chemins caractéristiques pour le test tout en restant automatisable.

Soient  $G$  un graphe représentant les comportements du système à tester,  $D$  une description de ce système,  $C$  un critère de couverture,  $E_C(D)$  un ensemble d'éléments de  $D$  satisfaisant  $C$  et  $N$  un nombre de tests à produire.

**Génération de chemins** La première étape de mon approche consiste à tirer un ensemble de  $N$  chemins tel que la probabilité minimale qu'a un élément de  $E_C(D)$  d'être atteint soit maximale. Si  $E_C(D)$  dénote directement un ensemble fini de chemins de  $G$ , comme dans le cas des critères du type "tous les chemins ...", la qualité de test est optimisée si les chemins de  $E_C(D)$  sont tirés uniformément. Par contre, si  $E_C(D)$  n'est pas décrit comme un ensemble de chemins mais un ensemble d'éléments constitutifs de  $G$  (des sommets, des arcs, des circuits, etc.) alors la probabilité minimale d'atteindre un élément lors du tirage des chemins doit être maximisée, car la génération uniforme de chemins sur laquelle je m'appuie ne le garantit pas. Le tirage des chemins devient plus compliqué et je décompose ce tirage en deux étapes. D'abord,  $N$  éléments  $e_1, \dots, e_N$  de  $E_C(D)$  sont choisis aléatoirement selon une distribution adéquate des probabilités. Puis, pour chaque élément  $e_i$  tiré, un chemin de  $G$  parmi tous ceux passant par  $e_i$  est tiré de manière uniforme.

Au cours de mes travaux, j'ai examiné trois distributions différentes sur les éléments : une distribution uniforme, une distribution basée sur la notion de "dominance" dans un graphe et une distribution déterminée par la résolution d'un système d'équations qui permet de maximiser la probabilité minimale d'atteindre un élément.

**Des chemins aux entrées** Une fois l'ensemble des  $N$  chemins obtenu, l'étape suivante consiste à déduire les valeurs d'entrées qui permettront de parcourir exactement ces chemins lorsque cela est possible (certains chemins peuvent ne correspondre à aucune exécution). On construit pour chaque chemin le prédicat qui décrit les relations entre les valeurs d'entrée qui permettent d'exécuter ce chemin, et on tente de résoudre ce prédicat. Dans le cas général, la résolution de prédicat est un problème indécidable. On s'appuie alors sur des

techniques de résolution de contraintes pour essayer de le faire. L'utilisation d'un solveur de contraintes randomisé est dans ce cas doublement avantageuse. Dans le cas où la résolution d'un prédicat est inconclusive (le solveur a atteint ses limites en temps passé à résoudre le prédicat et en nombre d'essais effectués), si on essaie de résoudre à nouveau ce prédicat, ce type de solveur permet de conclure plus facilement car les branches essayées et les valeurs choisies sont choisies aléatoirement et non dans un ordre déterminé. De plus, si un chemin est tiré plusieurs fois, la résolution randomisée permet d'obtenir différentes valeurs d'entrée.

**Application au test statistique structurel** Une première application a été effectuée pour le test statistique structurel défini par Thévenod-Fosse et Waeselynck (LAAS) et un prototype a été développé. La méthode du LAAS est basée sur la construction d'une distribution du domaine des entrées qui nécessite la résolution d'un système d'équations qui a autant d'équations que de chemins. Par conséquent, s'il y a un nombre infini ou trop important de chemins, cette construction doit être effectuée de manière empirique, ce qui freine son automatiser et donc sa généralisation. Les distributions utilisées par ma méthode et celle du LAAS n'étant pas de même nature, je ne peux comparer ces approches qu'expérimentalement.

**Évaluation expérimentale** J'ai repris plus de 10000 exécutions effectuées sur 2914 mutants, créés à partir de quatre fonctions issues d'un logiciel industriel et fournis par Thévenod-Fosse et Waeselynck. Ces expériences ont également servi à évaluer la stabilité du pouvoir de détection des fautes de ma méthode et à estimer si un passage à l'échelle était possible. Ces expériences montrent que mon approche est comparable à celle du LAAS, est stable et a l'avantage d'être complètement automatisée. Ces premières expériences me permettent également d'envisager un passage à l'échelle de mon approche.

#### 10.2.4 Conclusion

Ces travaux montrent qu'il est possible d'automatiser le test statistique structurel dont l'intérêt a déjà été démontré par Thévenod-Fosse et Waeselynck. Plus généralement, ma thèse montre qu'en combinant différentes techniques à première vue sans aucun rapport entre elles, on peut obtenir de nouvelles techniques comme ici une nouvelle approche du test statistique. De plus, ces travaux pourraient servir de base pour une nouvelle classe d'outils dans le domaine du test de logiciel, combinant génération aléatoire de structures combinatoires, techniques de programmation linéaire et résolution de contraintes.

## 11 Projet de recherche

### 11.1 Perspectives de recherche doctorale

Dans ma thèse, j'ai proposé une approche pour le test statistique basée sur la description sous forme de graphe des comportements du système à tester. Une première validation

expérimentale a été effectuée en appliquant cette approche au test statistique structurel. Afin de pouvoir illustrer l'application de cette approche à toute méthode de test, j'envisage une étude du test statistique fonctionnel et l'extension de mon prototype à de nouvelles descriptions (toujours sous forme de graphe) et à d'autres critères (par exemple définis par l'utilisateur).

### 11.1.1 Mieux exploiter les techniques combinatoires

Parallèlement, je compte étudier les moyens d'exploiter mieux les techniques combinatoires. La première approche consiste à les utiliser pour avoir une représentation plus précise des chemins de la description du système à tester en prenant en compte les propriétés structurelles ou certaines propriétés sémantiques de cette description. Ainsi, par exemple, la structure combinatoire pourrait exprimer la propriété suivante : "si je passe  $k$  fois dans la boucle A, alors je dois passer  $k$  fois dans la boucle B" afin de minimiser la génération de chemins infaisables.

La génération aléatoire des structures combinatoires se fait actuellement de manière uniforme. Si l'uniformité nous permet d'atteindre des éléments peu probables, elle ne rend pas forcément compte du profil opérationnel du système. Ainsi par exemple, les cas rares comme les cas d'exception sont testés avec la même intensité que le reste du code. Une approche consiste à prendre en compte le profil opérationnel tout en assurant une bonne couverture des éléments peu probables. Une solution serait alors de pouvoir orienter la distribution en fonction de métriques sur le programme ou d'indications sémantiques en mettant des poids sur les atomes de la spécification combinatoire (autre que 0 ou 1) ou sur les règles. Lors de notre dernière rencontre, j'avais proposé à Nicolas Thiery, responsable du groupe de développeurs de l'outil MuPAD-combinat, de pouvoir définir dans un premier temps les poids des atomes. Dans sa dernière version, l'outil MuPAD-combinat intègre cette option. L'étude de l'orientation de la distribution fait donc partie de mes perspectives.

Les outils actuels de génération aléatoire de structures combinatoires imposent de fixer une taille maximale des objets à générer. En pratique, le tirage se fait donc dans un ensemble fini de chemins de longueur bornée. Il serait intéressant d'étudier la possibilité de ne pas borner la taille des chemins en exploitant par exemple les résultats théoriques obtenus par la génération aléatoire basée sur le modèle de Boltzmann.

### 11.1.2 Limiter l'influence des chemins infaisables

Une autre perspective de recherche est le problème des chemins infaisables. Dans ma thèse, j'ai montré que la présence de chemins infaisables dans le graphe biaisait la distribution utilisée et faussait la qualité de test calculée lorsque les éléments du critère de couverture n'étaient pas des chemins. Il me paraît donc important d'étudier ce problème.

Indépendamment de l'amélioration des structures combinatoires manipulées, il est possible d'utiliser d'autres techniques pour réduire le problème des chemins infaisables. Parmi ces techniques, je m'intéresse aux techniques d'inférence et aux techniques probabilistes.

Concernant les techniques d'inférence, je compte continuer les travaux que j'ai commencés avec Nicolas Baskiotis et Michèle Sebag de l'équipe "Inférence et Apprentissage" du LRI. L'idée est de pouvoir identifier des motifs caractérisant les chemins infaisables à partir d'un ensemble de chemins identifiés comme étant faisables, infaisables ou indéterminés (cas où le solveur n'a pas trouvé de solution en temps fini) et en utilisant les techniques d'inférence et d'apprentissage. Les résultats de la première phase de l'étude permettent actuellement de pouvoir repérer les techniques les plus adaptées aux caractéristiques de nos chemins et de définir une modélisation des chemins qui permet à ces techniques d'être efficace (par exemple, en supprimant l'information redondante qui peut occasionner du bruit dans l'apprentissage).

Concernant l'approche basée sur les techniques probabilistes, j'aimerais étudier la possibilité de rajouter lors du calcul de la distribution un taux d'erreur (exprimant par exemple la probabilité qu'un chemin soit infaisable). Pour cette étude, il me faut trouver un partenariat avec des chercheurs du domaine probabiliste : un premier contact a déjà été établi avec des mathématiciens de Dijon au printemps 2003.

### 11.1.3 Autres

D'autres perspectives moins développées consistent à étudier les différentes applications possibles de mes travaux, comme par exemple le model-checking, ou encore l'utilisation des chaînes de Markov.

Je compte également retravailler sur mon outil AuGuSTe afin de pouvoir le diffuser dans la communauté du test.

## 11.2 Perspectives de recherche post-doctorale

Dans le cadre de ma recherche post-doctorale, j'étudie la génération de tests à partir de spécifications formelles JSL. Ces spécifications sont très proches des définitions que l'on utilise pour coder les types et les fonctions dans les langages fonctionnels tels que les langages ML. Ainsi à partir de ces travaux, j'entrevois au moins deux perspectives possibles.

### 11.2.1 Utilisation des structures combinatoires

La première consiste à utiliser les structures combinatoires pour la génération automatique de données. En effet, les types de données tels qu'ils sont définis me paraissent facilement modélisables par une structure combinatoire. L'utilisation des outils disponibles permettrait donc une génération uniforme et automatique de telles données. Le point délicat est la gestion du polymorphisme qui à mon avis demande une étude importante. Cette approche s'insère parfaitement dans le cadre du projet CASTLES : je commencerai donc cette étude cette année pour la continuer par la suite.

### **11.2.2 Extension aux langages fonctionnels**

La seconde consiste à étendre les travaux que je vais effectuer au cours de cette année au test des programmes écrits en langage fonctionnel (en prenant CAML ou Haskell comme point de départ) et pour lesquels il n'existe (à ma connaissance) ni méthode de test ni outil hormis le test aléatoire de propriétés de programme Haskell proposé par l'outil QuickCheck (travaux de Claessen et Hughes). L'intérêt de cette approche est double : d'abord cela permettrait de compléter la vérification (les programmes écrits en langage fonctionnel sont généralement facilement prouvables) par une validation. Ensuite, de récents travaux tentent de combiner tests et preuves soit en faisant du test à partir de preuves partielles (travaux de Lussier et Waeselynck) soit en testant des conjectures avant d'entamer des preuves souvent longues et compliquées (travaux de Berghofer et Nipkow).

### **11.3 Perspectives générales**

Les perspectives précédentes sont celles que je peux entrevoir à ce jour dans la ligne directe de mes travaux et des équipes dans lesquelles je me suis intégrée. Plus généralement, les thèmes qui m'intéresseraient sont le test de logiciel, le génie logiciel et les méthodes formelles, la validation et la vérification, la résolution de contraintes, la génération aléatoire et les structures combinatoires.

## **12 Participation à des groupes de travail**

### **12.1 Projet RNTL CASTLES (2004)**

Dans le cadre de ma recherche post-doctorale (cf. p.21), je participe à temps complet au projet CASTLES sur la Conception d'Analyses Statiques et de Tests pour le Logiciel Embarqué Sécurisé. Le projet Everest de l'INRIA Sophia-Antipolis, le projet LANDE de l'IRISA (Rennes) et les partenaires industriels AQL et Oberthur collaborent également à ce projet.

### **12.2 Projet national sur le Test de Robustesse (2001-2002)**

Au cours de ma thèse, j'ai participé à un projet national sur le Test de Robustesse (AS STIC No 23) animé par Richard Castanet et Hélène Waeselynck et dont le but était d'étudier les différentes faces de ce problème en fonction du type de système considéré, de déterminer le champ d'application des solutions existantes ainsi que les points durs nécessitant le développement de nouvelles approches. Ces travaux ont regroupé les équipes de cinq laboratoires : le LAAS (Toulouse) avec J. Arlat, O. Abdellatif-Kaddour et H. Waeselynck, le LABRI (Bordeaux) avec D. Janin et R. Castanet, l'IRISA (Rennes) avec C. Jard, T. Jérôme et H. Marchand, le LRI (Orsay) avec M.-C. Gaudel, G. Lestiennes et S.-D. Gouraud et le VERIMAG (Grenoble) : J.-C. Fernandez, L. Mounier et C. Pachon. Dans

ce projet, j'ai été amenée à faire des présentations et à écrire une partie du document de synthèse.

## 13 Exposés et participation à des colloques

### 13.1 Exposés à des conférences

2004

- Exposé à la cinquième conférence internationale IEEE *International Symposium on Software Reliability Engineering* (ISSRE), Saint Malo
- Exposé à l'atelier *Approches Formelles dans l'Assistance au Développement de Logiciels* (AFADL), Besançon
- Exposé au *MuPAD-Combinat first Developers and Users Meeting*, Rouen

2001

- Exposé à la sixième conférence internationale IEEE *Automated Software Engineering* (ASE), San Diego (USA).
- Exposé à l'atelier *AFADL*, Nancy.
- Exposé aux rencontres d'ALÉA, Marseille.

### 13.2 Autres participations à des conférences

2002 Participation à l'*International Symposium on Software Testing and Analysis* (ISSTA), Rome (Italie).

Lauréate d'une des bourses d'inscription financées par Microsoft.

## 14 Séminaires

**février 2005** Séminaire du laboratoire de l'IRISA, Rennes.

**janvier 2005** Séminaire du laboratoire Institut Gaspard-Monge, Marne-La-Vallée.

**janvier 2003** Séminaire d'équipe du projet Lande à l'IRISA, Rennes.

## 15 Développement

### 15.1 JSL2CHR

JSL2CHR est un module Prolog de 500 lignes que j'ai développé pour traduire automatiquement des spécifications JSL en des contraintes CHR. Ce module a déjà servi pour traduire les 310 fonctions décrivant la machine virtuelle Java Card en les 1537 contraintes CHR correspondantes.

## 15.2 AuGuSTe (*Automated Generation of Statistical Testing*)

AuGuSTe est un outil basé sur mes travaux de thèse que j'ai entièrement développé pour générer automatiquement des tests statistiques structurels. Il est essentiellement écrit en OCaml (environ 8000 lignes) et utilise des modules développés en C, Prolog, MuPAD et Java. Les modules C et MuPAD sont générés automatiquement par AuGuSTe. Les modules Prolog sont une adaptation du noyau de GATEL développé par Bruno Marre. Les modules Java sont des modules issus d'une librairie disponible gratuitement sur le Web que j'ai adapté à mon contexte.

## 15.3 SiRéMa (*Simulateur de Réseau Manhattan*)

SiRéMa est un outil de simulation et de visualisation d'un réseau Manhattan de communications, développé en Java (environ 10000 lignes). Il a été conçu, développé puis amélioré avec Antoine Barache et Allan Guillou lors d'un TER *Génie Logiciel* de licence puis un TER stage de maîtrise encadrés par Alain Denise. Lors d'un stage de maîtrise (été 1999), j'ai achevé cet outil en corrigeant ces bugs, en le mettant à jour par rapport à la nouvelle version de Java (version 1.1.4 vers 1.2) et en intégrant les dernières modifications demandées par les futurs utilisateurs Dominique Barth et Pascal Berthomé ainsi que la nouvelle interface graphique développée par Emmanuel Coquery.  
<http://www.lri.fr/~gouraud/sirema/sirema.html>



# AUTRES

## 16 Formations complémentaires

### 16.1 École Doctorale

Dans le cadre de l'école doctorale, les doctorants de première et deuxième année doivent suivre obligatoirement deux modules d'ouvertures scientifiques et professionnelles par an.

#### 2001-2002

- Génération aléatoire d'objets combinatoires et applications (A. Denise et D. Gouyou-Beauchamps, option du DEA Algorithmique)
- Modules du CIES

#### 2000-2001

- Spécifications Algébriques (M. Bidoit, option du DEA Programmation)
- Modules du CIES

### 16.2 Centre d'Initiation à l'Enseignement Supérieur (CIES)

Dans mon contrat de monitrice, il était stipulé que je devais suivre des jours de formation proposés par le CIES ou par les moniteurs eux-mêmes.

#### 2002-2003

- Intervention hebdomadaire dans le cadre d'un projet parrainé informatique (cf. p17)
- Recrutement et carrières

#### 2001-2002

- Intervention hebdomadaire dans le cadre d'un projet parrainé informatique (cf. p17)
- Responsabilité des enseignants (affaires juridiques) et Sécurité
- Systèmes étrangers (études et recherche)

## 2000-2001

- Apprendre à enseigner, à chercher, à communiquer (Alain Finkel)
- Lycée, DEUG et débouchés
- Métier d'enseignant et formation d'acteur
- Recherche et Protection Industrielle
- Une approche de la pédagogie et de la didactique des enseignements universitaires (Marc Legrand)

## 17 Expériences professionnelles

Mon premier contrat chez Renault République a été obtenu grâce à mon père, mécanicien dans ce garage, et au fait que je demandais une période de deux mois. Satisfaite de mon travail, la direction de Renault République a ensuite renouvelé mon contrat par deux fois et m'a donné en 1998 un poste de responsabilité puisque j'ai remplacé à temps complet pendant un mois leur comptable.

**Été 1998** Stagiaire vacances à la comptabilité générale (centralisation des caisses et fonds de caisse)

Succursale Renault République, 96-100, rue Amelot, 75011 Paris

**Été 1997** Stagiaire vacances à l'accueil et au standard téléphonique

Succursale Renault République, 75011 Paris

**Été 1996** Stagiaire vacances à la gestion du magasin des pièces de rechange

Succursale Renault République, 75011 Paris

**Juillet 1995** Stagiaire vacances au service comptabilité (chèques)

Crédit d'Équipement des Petites et Moyennes Entreprises

27-31, avenue du Général Leclerc, 94700 Maisons Alfort

## 18 Activités culturelles et sportives

Pendant une dizaine d'années (de 1984 à 1996), j'ai suivi des cours de solfège et de guitare classique au Conservatoire Georges Bizet, 54 rue des Cendriers, 75020 Paris. J'ai acquis un bon niveau : ce qui m'a permis de jouer de la musique de chambre en duo avec une flûte traversière pendant deux ans (1994 à 1996) et qui me permet aujourd'hui d'initier mon mari à la guitare.

Je pratique également des sports dits de la "forme" comme le STEP et le FAC, mais aussi le badminton et le ski alpin.