

Notice des titres et travaux

Février 2003,

*Claude JARD,
Directeur de recherche CNRS,
Détaché professeur à l'ENS Cachan-Bretagne*

I. BREF CURRICULUM VITAE

I.1. Etat civil

JARD Claude

Né le 12 Avril 1959 à Paris (17).
Nationalité française.
Exempté des obligations militaires.
Marié,
trois enfants nés le 17-02-83, 30-05-85 et 14-12-90.

Situation actuelle :

Directeur de recherche CNRS de 2^{ème} classe, 6 ans d'ancienneté dans ce grade, détaché professeur à l'ENS de Cachan.

Adresses professionnelles :

- IRISA, CNRS UMR 6074, Campus de Beaulieu, 35042 Rennes Cedex. Tél. 02.99.84.71.93, Fax. 02.99.84.25.32, Sec. 02.99.84.74.03, Claude.Jard@irisa.fr, http://www.irisa.fr/triskell/perso_pro/jard/accueil.html
- ENS Cachan-Bretagne, Campus de Ker-Lann, 35170 Bruz Cedex. Tél. 02.99.05.52.84, Claude.Jard@bretagne.ens-cachan.fr, <http://www.bretagne.ens-cachan.fr/DIT/People/Claude.Jard/accueil.html>

I.2. Cursus

Etudes et diplômes

- 1976-1978 : Mathématiques Supérieures et Spéciales au lycée Saint Louis, Paris.
- 1978-1981 : Elève ingénieur civil de l'Ecole Nationale Supérieure des Télécommunications de Bretagne (Brest).
- 1981 : Ingénieur des Télécommunications.
- 1984 : Thèse de Docteur-Ingénieur de l'université de Rennes1.
Titre : *Protocoles et Services – le test des spécifications*.
Jury : J.P. Verjus (président), P. Azéma, M. Raynal (rapporteurs), J.P. Ansart, Gv. Bochmann et J.M. Pitié.
- 1994 : Habilitation à diriger les recherches.
Titre : *Contribution à la vérification dynamique des protocoles*
Jury : J.P. Banâtre (président), Gv. Bochmann, JM. Pitié, J. Sifakis (rapporteurs), F. André, M. Raynal et J.P. Verjus.
- Qualifié sur la liste des professeurs en 1995.

Carrière professionnelle

- Fév.80-Sept.85 : Ingénieur-Chercheur au Centre National d'Etudes des Télécommunications (CNET) à Lannion, membre de l'équipe *Evaluation et Validation de Protocoles*. En 1981, j'ai passé une petite année dans l'équipe de Gv. Bochmann à l'université de Montréal comme chercheur invité.
- Oct.85-Sept.86 : Enseignant associé à l'université de Rennes1.
- Oct.86-Mars 97 : Chargé de Recherche (CR1) au CNRS, affecté à l'IRISA.
- Depuis Avril 1997 : Directeur de Recherche (DR2) au CNRS.
- Oct.86-Nov.95 : membre du projet INRIA *Algorithmes Distribués et Protocoles* jusqu'en 1991, puis du projet PAMPA (*Programmation des Architectures Massivement PARallèles*).
- Déc.95-Jui.01 : j'assure la direction de ce projet PAMPA comprenant en 2000 l'activité de plus de 25 personnes, dont 1 professeur (JM. Jézéquel), 3 Maîtres de Conférences (C. Viho, Y. Le Traon, N. Plouzeau), 3 chercheurs INRIA (T. Jéron, B. Caillaud, V. Rusu), une dizaine de doctorants, quelques post-doctorants et 6 ingénieurs en CDD. J'ai centré l'activité scientifique de ce nouveau projet PAMPA sur *les modèles et outils pour la programmation des architectures parallèles réparties*.
- En juillet 2001, ce projet s'arrête et contribue fortement à la naissance de nouvelles équipes en génie logiciel à l'Irisa, autour de JM. Jézéquel, T. Jéron et B. Caillaud.
- De avril 1999 à avril 2002, membre de l'équipe de direction de l'Irisa.
- Depuis septembre 2002, professeur dans le département Informatique et Télécommunications de l'ENS Cachan.

I.3. Responsabilités et travaux

Responsabilités scientifiques

- Membre du Comité National de la Recherche Scientifique (1991 - 1995) (*à ce titre, membre du comité de programme du P.I.R. Ville, membre des conseils scientifiques du LGI/Imag, de Bull/Imag et du LIP/Ens Lyon*).
- De 1993 à 1998, membre du comité de pilotage de *l'action Nouvelles technologies et la sécurité dans la ville* soutenue par l'IHESI et le CNRS.
- Membre du Comité des Projets de l'IRISA (depuis 1990).
- Membre du Conseil de laboratoire IRISA (de 1990 à 1998).
- Membre des commissions de spécialistes de l'université de Rennes, Brest et Orsay (de 1991 à 1995). Membre de la commission de spécialistes de l'université de Rennes (de 1998 à 2002) et de l'INSA (depuis 1998).
- Président de la commission personnel de l'Irisa (jusqu'en Avril 2002), commission qui instruit scientifiquement les dossiers de recrutement des personnels scientifiques temporaires (doctorants, post-doctorants, invités, ingénieurs-experts, ingénieurs associés, délégations, détachements). Soit environ 150 dossiers par an. C'est à ce titre que j'ai fait partie de l'équipe de direction du laboratoire.
- Membre du comité d'évaluation du laboratoire Vérimag.
- Membre du bureau du Comité National de la Recherche Scientifique (2000-2004) (*à ce titre, membre du comité d'évaluation du LAAS, de l'IMAG/LSR, de l'IMAG/ID, du LIRMM, du PRISM, de l'IGM*).
- Membre du conseil scientifique du département STIC (Science et Technologie de l'Information et de la Communication) du CNRS depuis septembre 2001.
- Membre du conseil stratégique de la société TNI-Valiosys.
- Membre du conseil scientifique du centre de recherche en informatique de Montréal.
- Membre du conseil scientifique de la revue Annals of Telecom.
- Membre du comité de pilotage du programme CNRS/STIC sur les systèmes embarqués.

Travaux de recherche

Mes travaux de recherche portent sur l'analyse des systèmes parallèles asynchrones. Ils s'inscrivent dans le cadre général de l'utilisation de méthodes formelles pour la programmation des architectures parallèles et réparties, et concernent les étapes de spécification, vérification et test des logiciels répartis sur des réseaux de processeurs. Le thème central de mes travaux est l'étude de méthodes de vérification dynamiques, dans lesquels la vérification est effectuée au cours de l'exécution - abstraite, simulée ou réelle - du programme à analyser.

Résultats : je suis l'auteur ou co-auteur de plus de 110 publications, dont 20 dans des revues internationales, 7 dans des revues nationales, 90 dans des colloques internationaux et nationaux (25 comme conférencier invité). J'ai co-édité 12 ouvrages. Ces publications se sont effectuées essentiellement au sein de trois communautés de recherche : l'informatique théorique, l'ingénierie des protocoles, et les systèmes répartis. J'ai participé à la conception et la réalisation de trois gros logiciels : le premier (Veda) a été pendant 10 ans au catalogue d'un industriel (Verilog); le second a été diffusé largement dans le milieu académique et a été évalué dans plusieurs sites industriels (Echidna), le troisième (TGV) a été transféré à la société Telelogic, il y a 2 ans.

Direction de recherches : j'ai encadré (ou co-encadré) 18 thèses de doctorat (2 en cours), participé à 68 jurys de thèse ou d'habilitation (34 en qualité de rapporteur).

Contrats : j'ai été à l'initiative et responsable scientifique de 15 contrats, 10 avec des industriels, et 5 conventions de coopération internationale (Canada, Israël et Europe). Depuis plusieurs années, ces contrats rapportent au laboratoire environ 500kF par an.

Comités de rédaction et comités de programme : j'ai été membre du comité de rédaction de 2 revues nationales et ai édité plusieurs éditions spéciales, membre du conseil scientifique des annales des télécommunications. J'ai été membre du comité de programme de 17 conférences internationales (26 conférences nationales) et ai (co-)organisé plusieurs colloques nationaux, 5 écoles d'été et 4 congrès internationaux. Activité régulière de rapporteur pour quelques revues internationales et de nombreux colloques. Je suis rapporteur attitré pour IEEE Computer. Des expertises ont été aussi réalisées à la demande de la CEE (programmes européens), de ministères (appels d'offres informatique et automatique, ANRT, NSERC Canadien, RNRT, RNTL) ou des tutelles (CNRS, INRIA).

Enseignement : je participe depuis 1984 à l'enseignement de 3ème cycle de l'université de Rennes1. Je suis responsable d'un module de cours du DEA informatique (depuis 1988), d'un module dans l'école d'ingénieur de l'université (depuis 1993) et dans l'école des télécommunications. L'ensemble représente une moyenne annuelle de 50 heures de cours. En 1986, j'ai assuré un service complet d'enseignement, y compris en second cycle. Dans l'année universitaire 2001-2002, j'ai assuré un enseignement d'informatique en première année du magistère Math-Info (Univ. Rennes 1 / ENS Cachan). Cette année, j'assure un service complet d'enseignement à l'ENS (dans les magistères MIT et Math).

II. ENSEIGNEMENT

II.1. 3^{ème} cycle

Depuis 1988, j'ai la responsabilité de modules de cours (faisant participer d'autres chercheurs et enseignants)- dans le DEA informatique de l'université de Rennes (fondements de la programmation distribuée), dans le diplôme d'ingénieur de l'université de Rennes (DIIC3) (validation de protocoles) et à l'école supérieure des Télécommunications de Rennes (validation de protocoles).

En moyenne, mon activité d'enseignement représente environ un volume de 50 heures de cours par an.

Le contenu des cours est composé essentiellement de mon expérience en validation de protocoles. J'y présente les différentes techniques et les algorithmes associés. J'interviens donc essentiellement sur deux thèmes : l'algorithmique répartie et le génie logiciel (vérification et test). Dans quelques formations, j'ai conçu des travaux pratiques, fondés sur l'utilisation des outils Veda, et Echidna et ObjectGéode. Plus précisément, le cours présente plusieurs parties : une introduction à la problématique de la vérification, l'algorithmique de la vérification à travers le "model-checking", la génération de test et le problème de l'observation en univers réparti avec son algorithmique associée. Il s'agit donc d'un cours à profil assez "vertical" : les objets étudiés sont les protocoles et les algorithmes distribués, et les problèmes abordés vont de la description formelle à l'implantation. Les modèles utilisés pour présenter les techniques et exemples sont les automates communicants, les réseaux de Petri, les logiques temporelles et les ordres partiels. Plus récemment, j'ai investi dans l'enseignement du test et avec une ouverture sur la modélisation objet.

Actuellement, les travaux pratiques de validation de protocoles durent 15h, et sont effectués par petits groupes en utilisant l'outil industriel Object-Géode. Le protocole de connexion-déconnexion sert d'abord d'illustration, puis le protocole du bit-alterné bien connu dans la communauté. Un aspect original du travail consiste à mettre les étudiants dans une situation réaliste d'observation répartie dans laquelle ils ont à spécifier des processus observateurs pour détecter en ligne des erreurs. Cet aspect a fait l'objet de publications dans des cercles pédagogiques comme la revue *Computer Science Education* [RI17,CI36].

II.2. Second cycle

En janvier 2002, j'ai pris en charge le cours de compilation dans le magistère Math et Informatique de l'ENS Cachan-Bretagne, dans l'objectif de préparer un détachement en tant que professeur dans cette école. Le cours de compilation de 1^{ère} année (niveau licence) est l'occasion de motiver un public d'élèves mathématiciens à l'informatique. Ils découvrent que l'informatique ne se résume pas à la programmation et qu'elle a ses propres modèles, fondements et questionnements. C'est aussi le lieu où l'on peut effectuer quelques digressions pour leur faire sentir la complexité (et la beauté quelquefois) des machines informatiques. Etant donné la vocation des magistères à former par la recherche et pour la recherche, je me suis aussi occupé de trouver et suivre des stages d'été de recherche en informatique pour ces élèves du cours de compilation. Le résultat le plus concret est que 3 d'entre eux ont choisi de continuer leur formation en informatique, plutôt qu'en mathématiques.

Depuis la rentrée 2002-2003, je suis "en poste" à l'ENS et assure pleinement les charges d'enseignement et de suivi des élèves qui en découlent. Je suis au sein d'une petite équipe pédagogique comportant 2 professeurs (Luc Bougé, directeur du département, et moi-même), 1 maître de conférences (David Cachera) et 2 moniteurs. Je suis responsable du cours sur les réseaux (1^{ère} année du magistère Informatique et Télécom : MIT) et du cours de compilation (1^{ère} année du magistère Math-Info). J'assure aussi les TD du cours d'algorithmique des graphes donné par Luc Bougé. L'autre aspect de ma tâche d'enseignant est l'animation d'un séminaire à destination des élèves du département, le montage et le suivi des stages d'étude et de recherche, ainsi que le tutorat de quelques élèves.

II.3. Autres

En dehors des colloques, je suis invité de temps en temps à donner des séminaires dans les laboratoires académiques (ces dernières années : INT, LABRI, LIFC, Vérimag, quelques universités étrangères lors de visites ...) et industriels.

III. TRAVAUX DE RECHERCHE

III.1. Présentation et objectifs de la recherche

Le développement des réseaux d'ordinateurs permettant l'interconnexion de machines se poursuit. Aussi les questions posées par la construction et le suivi du logiciel pour ces systèmes sont d'une grande actualité. Même si des progrès spectaculaires ont été accomplis dans les méthodes de génie logiciel, le caractère intrinsèquement parallèle et réparti des logiciels mis en œuvre continue à poser des problèmes ardu. La question la plus sensible à mes yeux est celle de la maîtrise de la fiabilité du logiciel, c'est-à-dire le contrôle des conditions de son bon fonctionnement. Cette préoccupation passe par le renforcement des activités de conception, vérification, test et supervision. Mes recherches continuent à explorer ce spectre, en allant vers des systèmes de plus en plus complexes et des techniques qui peuvent être déployées à grande échelle, comme la validation répartie « en ligne » des applications réparties sur des réseaux.

Mon ambition est de continuer à contribuer à l'élaboration de nouvelles technologies logicielles par l'étude de modèles formels des protocoles et l'invention d'outils informatiques associés. Je privilégie la conception d'outils automatiques permettant d'aider aux tâches de conception, vérification et test de programmes réels. Ces outils ont vocation à être diffusés dans le milieu académique et/ou industriel. La mise au point des modèles et outils s'effectue dans le cadre d'applications réparties situées principalement dans le domaine du logiciel pour les télécommunications.

Depuis plusieurs années, je me concentre (avec tout un groupe de chercheurs) sur les techniques par modèles (dites « model-checking »), et particulièrement sur les aspects algorithmiques (algorithmique à la volée). Nous avons étendu ces techniques pour être capables de générer automatiquement des séquences de test de conformité à partir de spécifications dans des langages comme SDL ou Lotos. Cela a conduit à un outil original par son algorithmique et son architecture (appelé TGV), que nous avons valorisé auprès de la société Telelogic. Un autre aspect du test est l'évaluation des comportements d'un code réparti à partir des traces qu'il produit. Pour modéliser les phénomènes de causalité et de concurrence, la théorie de l'ordre est notre outil mathématique de base. Nous avons plusieurs applications comme le test d'interopérabilité et le diagnostic de pannes dans les réseaux. Enfin, pour qu'elle ait un véritable impact, la validation (vérification/test) doit s'intégrer dans des environnements et méthodes de conception existantes. Pour cela, le paradigme objet et la notation UML sont maintenant incontournables, et mes recherches se situent essentiellement maintenant dans le contexte de l'assemblage fiable de composants logiciels à l'aide d'un environnement de conception UML outillé avec des méthodes formelles, et dans le cadre des grands systèmes répartis.

III.2. Domaine d'application

Mon activité de recherche, centrée sur la maîtrise de la fiabilité, est en rapport avec deux types d'applications dans le domaine des télécommunications : la conception fiable de logiciels communicants et le test et diagnostic de systèmes communicants.

Conception fiable de logiciels communicants

L'exigence de fiabilité des logiciels est facile à comprendre dans un contexte où ils sont présents à de très nombreux exemplaires et dans différentes versions sur un grand réseau de télécommunication. L'accent est porté sur la faculté d'interopérer. Le coût d'apparition d'une faute majeure est considérable dans ces systèmes et la réparation longue et difficile. Il est à noter aussi une exigence d'évolutivité importante liée à la mise en œuvre rapide de nouveaux services. Je crois à l'utilisation de méthodes formelles pour faciliter la résolution de ces problèmes. Mais il ne faut pas oublier que les méthodes formelles doivent de plus en plus s'intégrer à une « approche système » permettant aux ingénieurs de concevoir globalement les systèmes pour prendre en compte tout un ensemble de contraintes et d'objectifs liés aux besoins des utilisateurs. Ces méthodologies formalisées n'en sont qu'à leurs débuts : un bon exemple est l'approche objet qui s'étend rapidement au contexte des télécommunications.

Test et diagnostic des systèmes communicants

Le test est une autre facette du développement fiable ; il consiste à s'assurer que le système, une fois réalisé, est conforme à ses spécifications. Quel que soit le soin apporté à la conception, cette phase reste de première importance pour vérifier le bon fonctionnement du système dans des environnements complexes et évolutifs. La surveillance et

le diagnostic sont aussi des aspects du travail sur les implantations. Je me suis focalisé sur la génération automatique de tests de conformité à partir de spécifications formelles, ainsi que sur le diagnostic en environnement réparti. L'obtention d'une bonne suite de tests de conformité est d'un intérêt économique certain. C'est à l'heure actuelle un travail manuel coûteux et répétitif qui est ensuite utilisé à grande échelle. Nous participons au défi de l'automatisation en mettant particulièrement l'accent sur la qualité de la suite de tests (dans sa capacité à détecter les implantations non conformes et uniquement celles-ci). Sur plusieurs études de cas, nous avons montré la rentabilité de l'approche et le transfert industriel de notre outil TGV en est l'exemple. Je m'attache plus particulièrement aujourd'hui à étendre des techniques de génération pour produire des tests répartis contenant du parallélisme, afin d'aborder les questions de test d'interopérabilité entre composants.

III.3. Les fondements

III.3.1. Fondements mathématiques des systèmes réactifs et répartis

La structure mathématique qui caractérise le mieux les fondements des travaux de recherche en vérification et génération de tests de programmes répartis sont les systèmes de transitions étiquetés (« labelled transition systems » en anglais, abréviation LTS) [1]. Cette structure, développée il y a près de cinquante ans est l'un des fondements de l'informatique ; aussi il est utile de préciser de quelle façon nous utilisons cette structure, notamment sa construction au vol. L'autre aspect fondamental est la notion de causalité entre événements dans les exécutions réparties. C'est le concept central qui permet de parler de l'analyse des comportements des systèmes distribués [2]. Il est aussi à la base des plus beaux résultats en algorithmique répartie.

Systemes de transitions

Un LTS est un graphe orienté dont les arêtes, appelées transitions, sont étiquetées par une lettre prise dans un alphabet d'événements. Les sommets de ce graphe sont appelés états.

$M=(Q^M, A, T^M$ dans $Q^M \times A \times Q^M, q_{init}^M)$ avec : Q^M ensemble des états, q_{init}^M l'état initial, A l'ensemble des événements, T^M la relation de transition.

Il est usuel de parler d'automate d'états finis pour désigner un système de transitions étiqueté dont l'ensemble des états et celui des événements sont finis. Il s'agit en fait du modèle de machine le plus simple que l'on puisse imaginer. Nous employons les LTS pour modéliser des systèmes réactifs le plus souvent répartis. Dans ce cadre les événements représentent les interactions (entrées ou sorties) du système avec son environnement. On parle alors de système de transitions « entrées-sorties » ou de IOLTS (« input-output LTS »).

Ces systèmes de transitions sont obtenus à partir de spécifications de systèmes réactifs répartis décrits dans des langages de haut niveau comme SDL, Lotos et maintenant UML. L'association d'un LTS à un programme se fait par l'intermédiaire d'une définition opérationnelle de la sémantique du langage et est en général formalisée sous la forme d'un système de déductions. Pour un langage aussi simple qu'une algèbre de processus (CCS par exemple), la définition de sa

sémantique opérationnelle tient en moins de dix axiomes et règles d'inférences ; alors que pour un langage aussi complexe que SDL, cela est plutôt l'affaire d'un document de plus de cent pages ; quant à UML, le problème est particulièrement ardu du à l'informalité de plusieurs zones de la notation.

Pour des raisons de performance, ces sémantiques opérationnelles ne sont jamais mises en œuvre directement ; mais font l'objet de transformations diverses. En particulier, la compacité du codage des états est un facteur déterminant de l'efficacité de la génération des LTS. Les calculs et transformations opérés sur les LTS se résument à des parcours et calculs de points fixes sur les graphes. L'originalité réside dans la façon de les effectuer : par calcul explicite du LTS ou bien implicitement, sans calcul ou stockage exhaustif du LTS. Les algorithmes classiques de théorie des langages construisent explicitement des automates d'états finis. Ils sont le plus souvent intégralement stockés en mémoire. Cependant, pour les problèmes qui nous intéressent, la construction (ou la mémorisation) exhaustive des LTS n'est pas toujours nécessaire. Une construction partielle suffit et des stratégies analogues aux évaluations paresseuses des programmes fonctionnels peuvent être employées : seule la partie nécessaire à l'algorithme est calculée. Dans le même esprit il est possible d'oublier certaines parties précédemment calculées du LTS ; et par recyclage judicieux de la mémoire, il est possible d'économiser l'espace mémoire utilisé par nos algorithmes. La combinaison de ces stratégies de calcul sur des LTS implicites permet de traiter des systèmes de taille réelle même en utilisant des

moyens de calcul tout à fait ordinaires. C'est un aspect auquel j'ai particulièrement contribué ces dix dernières années du point de vue des modèles, des algorithmes, de la mise en œuvre et de la valorisation.

Ensembles Partiellement Ordonnés

On considère qu'une exécution répartie sur un réseau de processus I est faite d'événements atomiques E , certains étant observables, d'autres ne l'étant pas. Chaque événement est l'occurrence d'une action ou opération (Notons S l'alphabet des actions) ; on considère habituellement qu'une action a lieu sur un seul et même processus du réseau. Nous avons donc :

- I un ensemble fini de processus
- S un ensemble d'actions
- $P : S$ vers I , un placement des actions
- E un ensemble d'événements
- $F : E$ vers S , un étiquetage des événements

La relation de causalité décrit le plus petit ordonnancement partiel sur les événements que l'on peut déduire du modèle de fonctionnement du réseau de processus que l'on s'est donné. Il a été présenté sous cette forme pour la première fois dans [3] avec comme hypothèses sur le fonctionnement de l'architecture répartie :

- Les processus sont séquentiels. Deux événements ayant eu lieu sur le même processus sont ordonnés.
- Les communications sont asynchrones et points à points. L'émission d'un message précède causalement sa réception.

Ces deux axiomes nous permettent de définir une relation d'ordre *prec* sur E : c'est la relation de causalité. Tout ce que l'on peut dire est que tout ordonnancement aurait respecté l'ordonnancement causal ; autrement dit, le comportement réel du système est une *extension linéaire* de l'ordre de causalité. Il n'est cependant ni réaliste ni même utile de chercher à savoir quelle extension linéaire s'est réellement produite. Cela n'est pas réaliste car les architectures réparties existantes n'offrent pas les moyens de synchroniser des horloges locales à chaque processus avec une précision suffisante. Cela n'est pas utile car cet ordonnancement dépend de conditions d'exécution qui ne sont pas contrôlables ou reproductibles. Il faut donc considérer que toute extension linéaire de la relation de causalité est un ordonnancement plausible.

La difficulté est dans la combinatoire en général exponentielle dans le nombre d'événements des extensions linéaires d'une relation d'ordre. Il existe cependant une structure intéressante pour représenter l'ensemble des états dans lequel le système a pu se trouver : c'est le treillis des antichaînes de la relation d'ordre [4]. En terme d'exécution répartie ceci correspond à désigner pour chaque processus quel a été le dernier événement qui s'est produit ; cela définit sans ambiguïté un état possible du système. Ce treillis (distributif) peut être représenté sous la forme d'un LTS, avec comme relation de transition, la relation de couverture. Il s'agit du treillis des états globaux.

Ce LTS est l'analogue du graphe d'état global pour les systèmes déterministes ne possédant qu'une seule exécution (d'ordre partiel). Pour capturer l'ensemble des ordres partiels d'un système général non déterministe, sans pour autant calculer tous les entrelacements, on utilise le modèle des *structures d'événements*. Dans ce modèle, les différents ordres partiels sont superposés. On les distingue néanmoins en considérant une nouvelle relation binaire sur les événements, appelée « conflit ». L'idée est que deux événements en conflit ne peuvent pas être présent dans la même histoire (ordre partiel) du système. Le conflit élimine donc des concurrences exprimées lors de la superposition des ordres. On considère qu'il est hérité par causalité, c'est à dire que si un événement est en conflit avec un deuxième, leurs successeurs causaux le sont aussi. Une telle structure d'événements est une représentation d'ordre partiel de l'ensemble des comportements du système et présente une alternative à la représentation par LTS ou automate. Le gain en taille est spectaculaire pour les systèmes à fort parallélisme. Elle a aussi le mérite d'explicitier le parallélisme et de ne pas confondre parallélisme et non déterminisme dans la structure du graphe. Elle a par contre le défaut de nécessiter des algorithmes plus complexes pour calculer des représentations finies.

III.3.2. Génération automatique de tests

Le test de conformité est un test de type boîte noire. On se donne une spécification d'un système ouvert qui sert de modèle de référence et une implantation réelle de ce système (appelée IUT pour « Implementation Under Test »)

dont on ne connaît le comportement que par ses interactions avec l'environnement (à travers des points de contrôle et observation, appelés PCO). Un test consiste à alterner le contrôle de l'IUT et son observation par un testeur et en déduire un verdict sur la conformité de l'IUT par rapport à sa spécification en fonction d'une relation de conformité. La génération automatique de test consiste à produire automatiquement des cas de tests en fonction de la spécification et de la relation de conformité choisie. La sélection d'un ensemble significatif de cas de tests peut se faire en utilisant des objectifs de test comme cela se fait lors de l'écriture manuelle des tests.

Le modèle de base permettant la modélisation des objets relatifs au test de conformité est le modèle IOLTS. Il distingue deux types d'actions, les actions internes inobservables, les actions observables elles-mêmes séparées en entrées et sorties, permettant ainsi de modéliser l'observation et le contrôle. La génération partant de la spécification et d'un objectif de test implique des transformations de modèles tels que l'abstraction d'actions internes, la déterminisation, la minimisation qui créent des instances de sous-classes du modèle général d'IOLTS. Une des relations qui lie les modèles d'IOLTS est la relation de conformité. La relation de conformité *ioco* choisie, définie par Jan Tretmans [5] (Université de Twente) stipule qu'une IUT I est conforme à la spécification S si après toute trace (séquence d'actions observables y compris les blocages) de la spécification, les sorties possibles de l'IUT (y compris les blocages) sont incluses dans celles de la spécification.

La génération de test implantée dans notre outil TGV utilise une algorithmique de graphe, en particulier des algorithmes de parcours [R116, RN6, CV21, CV22]. Ces parcours font aussi de la construction à la volée. TGV utilise plusieurs parcours en profondeur à plusieurs niveaux dans l'outil et en particulier des adaptations d'un algorithme de Tarjan [6]. Cet algorithme permet de calculer les composantes fortement connexes maximales (CFC) d'un graphe dirigé. L'algorithme original est récursif mais pour des raisons d'efficacité, nous utilisons une version itérative. Sa complexité est linéaire en temps et en mémoire. Par définition, les CFC sont des classes d'équivalence pour la relation d'accessibilité et de co-accessibilité: deux sommets u et v sont dans la même CFC si v est accessible depuis u et u accessible depuis v . On peut aussi remarquer que les CFC peuvent se caractériser par une autre relation : depuis tous les sommets d'une même composante on peut atteindre exactement les mêmes ensembles de sommets. Cette caractérisation permet de se rendre compte de l'utilité du calcul de CFC pour la vérification de propriétés d'accessibilité. C'est cette idée qui est utilisée pour adapter l'algorithme de Tarjan à la génération de tests car justement plusieurs problèmes se réduisent à des problèmes d'accessibilité. La première adaptation est utilisée pour calculer une tau-réduction (abstraction des actions internes). Le problème se traduit en l'accessibilité aux transitions visibles. De plus les boucles d'actions internes tau sont caractéristiques des divergences pour les LTS finis et ces divergences sont considérées comme observables dans la relation de conformité donc par un test. La seconde adaptation est utilisée dans l'algorithme principal de TGV. Ici, le problème est celui de l'accessibilité à un ensemble d'états puisque, pour schématiser, on veut calculer un sous-graphe du produit entre spécification et objectif de test contenant l'ensemble des états (ou seulement une partie) depuis lesquels les états accepteurs de l'objectif de test sont accessibles. Enfin, une troisième adaptation est utilisée pour extraire du sous-graphe précédant une partie dite « contrôlable » c'est à dire n'ayant jamais de choix entre une action contrôlable et d'autres actions (contrôlables ou non). De façon moins immédiate, ceci peut se voir aussi comme un problème d'accessibilité à l'état initial sur le graphe où on a inversé la relation de transition et modifié celle-ci au vol par coupure des conflits de contrôlabilité.

III.4. Nouveaux résultats obtenus ces dernières années

Ils concernent l'intégration de techniques de validation dans des environnements UML, la génération de test répartis (sous les deux aspects complémentaires du test asynchrone et de la distribution automatique de testeurs), la manipulation formelle de langages de scénarios et la supervision de systèmes répartis.

III.4.1. UML et « model-checking »

L'utilisation assez massive d'UML comme notation d'aide à la conception de composants logiciels et l'absence criante d'outils de validation associés doit interpeller fortement la communauté des méthodes formelles dont je fais partie. Aussi, nous avons choisi de travailler à l'idée que beaucoup de techniques formelles existantes pourraient être rendues disponibles aux utilisateurs d'UML. Du même coup, ces techniques acquièrent une intégration réelle dans le processus de conception. Nous nous sommes donc focalisés sur le model-checking au vol et la génération de tests à partir de graphes, qui forment nos compétences de fond. Dans le cadre de la thèse d'A. Le Guennec, nous avons donc montré comment réaliser cette intégration à partir du diagramme de classes, des state-charts (avec quelques

contraintes) et du diagramme de déploiement [RN7]. Nous avons ensuite approfondi la méthodologie en proposant de spécifier les objectifs de test en utilisant les diagrammes de séquences de UML [CI50].

III.4.2. Test asynchrone

L'architecture réelle sur laquelle les tests répartis sont effectués a une importance évidente pour la génération de tests : l'IUT est généralement enfouie dans un contexte et l'interaction avec le testeur n'est pas directe. Ainsi, l'IUT dans son contexte est conforme à une spécification dans le même contexte si la composition de l'IUT avec son contexte est conforme (hors contexte) avec la spécification dans le même contexte. Pour une spécification donnée, on constate malheureusement que les ensembles d'IUT conformes dans et hors contexte ne correspondent pas : le contexte rend le test plus laxiste (des IUT non conformes deviennent conformes dans un contexte) mais ce qui est plus gênant, des IUT conformes deviennent non conformes dans un contexte. Nous avons cependant montré que le deuxième problème pouvait être évité en rendant (ou considérant) les spécifications complètes en entrées et pour des contextes monotones (préservant l'inclusion de trace).

Dans le cas d'un contexte asynchrone réduit à un PCO (une file FIFO dans chaque sens), nous avons aussi montré qu'en instrumentant l'IUT d'un mécanisme simple de comptage d'événements observables, on pouvait également supprimer le problème du laxisme. En résumé, nous avons montré que le test distant asynchrone avec un PCO avait la même puissance d'observation (le contrôle est quand même réduit) que le test synchrone. Nous proposons deux techniques de génération/exécution de tests. La première consiste à générer les tests à partir de la spécification instrumentée et transformée par le contexte asynchrone. La deuxième consiste à générer les tests à partir de la spécification hors contexte et de les exécuter sur l'IUT instrumentée en utilisant un « driver » de test qui pilote le test et décode l'ordre d'occurrence des événements de l'IUT [CI37].

III.4.3. Distribution automatique de testeurs

Nous considérons le problème du test réparti dans lequel, dans le cadre le plus général, plusieurs testeurs communiquant entre eux de manière asynchrone interagissent de manière asynchrone avec une implantation elle-même répartie.

Nous avons tout d'abord défini ce qu'était un test séquentiel valide communiquant de manière asynchrone avec une implantation. A partir d'un tel test séquentiel, nous avons proposé un schéma de distribution permettant de produire un ensemble de tests communicant entre eux de manière asynchrone et s'appuyant sur un service global de consensus permettant de réaliser des choix non locaux. Nous avons prouvé que l'ensemble des tests ainsi produits avait le même pouvoir de détection d'erreur que le test séquentiel. Néanmoins ce schéma impose pour l'instant trop de synchronisation entre testeurs du fait que l'information de concurrence entre actions du testeur séquentiel n'a pas été conservée. C'est pourquoi j'ai poursuivi ce travail en utilisant des modèles de vraie concurrence (cf. 4.4.). Nous avons aussi étudié un autre schéma de distribution dans lequel les testeurs parallèles se synchronisent suivant un mécanisme de type round-robin [CI40]. Ceci permet d'optimiser les synchronisations et de traiter proprement la distribution des mécanismes des timers. Les algorithmes ont été mis en œuvre sur une grosse plate-forme de test du centre de recherche de l'armée [CO15].

Par ailleurs, nous avons montré que l'algorithmique de la synthèse de test pouvait être utilisée pour vérifier automatiquement la correction de cas de test écrits à la main, ce qui étend de façon pragmatique le champ d'utilisation de notre technique [CI41].

III.4.4. Génération de tests parallèles

Je me suis attaqué à la révision de la méthode TGV [CI47] en fondant la représentation des comportements de la spécification non plus sur des automates séquentiels, mais sur des modèles de vrai-parallélisme (les structures d'événements en fait, cf. 3.1.). Cela permet de synthétiser des cas de test parallèles utilisant le parallélisme intrinsèque de la spécification (test de systèmes multi-composants) et en définissant des objectifs de test eux-mêmes parallèles. Les principes sont publiés dans [RI19, CI45, CV24]. Un prototype a été réalisé intégrant le logiciel ERVUnfold de l'université de Munich et divers algorithmes d'abstraction et de projection des structures d'événements, réalisés par des stagiaires de l'Irisa.

III.4.5. Langages de scénarios

L'usage de scénarios pour spécifier les comportements de systèmes est une pratique bien établie. Des notations formelles ont été proposées comme les « High level Message Sequence Charts » (HMSC), principalement utilisés pour décrire l'activité de processus communicants dans le domaine des protocoles [CV25]. On les retrouve sous une forme proche, les diagrammes de séquence, dans une des vues définies par la notation UML.

Dans le cadre de la thèse de L. Hérouët, nous avons défini une sémantique d'ordre partiel pour des familles de scénarios représentés par des HMSC. Celle-ci permet d'associer à chaque HMSC une structure d'événements engendré par une grammaire de graphes finie [RI18]. On peut alors décider effectivement l'équivalence (l'isomorphisme des structures d'événements) de deux HMSC [CI38]. Nous avons développé des applications de ce résultat à la conception d'un simulateur de scénarios [CI43, CV23] et d'un générateur d'automates [CI39].

Nous avons cette année effectué une connexion intéressante entre le travail sur les scénarios et le travail sur les tests. Nous avons défini un nouveau langage de description de tests abstraits pour le contexte de test de composants logiciels. Ce langage a été appelé Tela [CI44]. L'originalité a consisté à se placer dans le cadre composants et objets, et à mettre l'accent sur les aspects comportementaux et d'ordre partiel. L'approche technique a été d'embarquer le plus possible le langage dans les diagrammes de séquences de la notation UML, afin de s'intégrer à une pratique industrielle. Le résultat est une solution mieux intégrée que les propositions usuelles d'habillage d'un langage de test existant avec UML. Notre proposition a d'ailleurs alimenté largement les travaux d'un groupe de travail de l'OMG, chargé de la définition d'un profil de test pour UML. Nous avons principalement utilisé les diagrammes de séquence, et avons dû discuter en profondeur des aspects sémantiques des scénarios, profitant de notre expertise sur les HMSC [RI18]. Tela est utilisé comme langage pivot dans la chaîne d'outils développée dans le projet COTE. Il constitue la notation cible pour Umlaut/TGV, alimenté par un objectif de test décrit dans une version simplifiée et abstraite de Tela (O-Tela). C'est aussi la notation source pour les générateurs de tests exécutables situés en aval (EJB, .Net et Corba).

III.4.6. Diagnostic réparti

Le cadre est celui des systèmes à événements discrets, répartis et asynchrones. Les réseaux de télécommunications, mais aussi les grandes architectures logicielles à base de composants sont des exemples de tels systèmes. Dans ceux-ci, on imagine que le système d'observation (les capteurs) est lui aussi réparti ; chaque capteur local n'ayant qu'une vue partielle de l'état global. Ils ne sont pas fortement synchronisés et les alarmes doivent être traitées de façon asynchrone.

C'est typiquement le cas des applications de télécommunication étudiées dans le projet Magda2.

Dans ce contexte où les alarmes se promènent sur le réseau, la vision d'une "séquence d'alarmes" n'est plus un ordre total, mais un ordre partiel. L'approche originale que nous développons travaille en fait sur des trajectoires que nous manipulons en tant que graphes. De plus, le futur de la supervision nous apparaît répartie, avec une notion de supervision locale dont les diagnostics doivent être maintenus cohérents (ils doivent être perçus comme des projections locales d'un diagnostic global que l'on ne cherche pas forcément à reconstruire). C'est ce que l'on a défini comme la problématique du diagnostic réparti.

Cette recherche est menée depuis plusieurs années en collaboration avec l'équipe Sigma2 (A. Benveniste, E. Fabre et S. Haar) et concerne les aspects modèles et algorithmes. Une technologie originale, fondée sur une notion nouvelle de Réseau de Petri stochastique conforme au point de vue dit de la « concurrence vraie » a été développée. Cette technologie nous permet naturellement de prendre en compte toutes les contraintes qui résultent du contexte distribué où nous nous situons. Nous utilisons un modèle de temps en « ordre partiel » : deux événements ne sont ordonnés que s'ils surviennent sur un même site, ou bien s'ils sont reliés par une relation de causalité (par exemple en tête et en queue d'une communication). Les techniques indiquées ci-après doivent être adaptées à ce point de vue, ce qui fait la difficulté du sujet [CI42, CI48].

Le réseau de Petri permet de décrire l'apparition spontanée de pannes ou dysfonctionnements, ainsi que leur propagation et effets induits. Il joue ainsi, dans un cadre dynamique et distribué, le rôle d'arbre de défaillances. Le franchissement d'une transition matérialise un changement d'état de (dys)fonctionnement et cause l'émission d'un faisceau d'alarmes observées de manière répartie sur le réseau. Les transitions et l'émission des alarmes peuvent être

randomisées (tout particulièrement la perte éventuelle des alarmes). Le diagnostic est alors effectué par le biais 1/ du calcul à la volée de l'ensemble des trajectoires d'état interne compatibles avec les alarmes observées (formalisé par un dépliage contraint du réseau de Petri [CI49]), et, en cas de multiplicité de solutions, 2/ le calcul de la trajectoire la plus probable par une technique de maximum de vraisemblance mise en œuvre via une programmation dynamique généralisée aux systèmes distribués (l'algorithme de Viterbi [CI46]). Le cadre formel des dépliages des réseaux de Petri permet de donner un cadre concret de représentation compacte des modèles, observations et trajectoires à reconstruire [RI20].

III.5. Projets à court terme

III.5.1. Vers une théorie de la conformité d'ordre partiel

Sous l'hypothèse que les causalités et la concurrence peuvent être observées lors de l'exécution du test (au prix d'une instrumentation des composants), on peut vérifier ainsi une relation de conformité « parallèle » plus fine que celle admise habituellement dans le domaine du test de conformité. Il s'agit d'établir cette théorie, en correspondance avec la théorie habituelle fondée sur les systèmes de transitions. Une expérimentation de la chaîne de synthèse de tests parallèle sur une application réaliste est en cours d'étude avec le laboratoire Vérimag à Grenoble.

III.5.2. Sémantique comportementale de UML à base d'ordres partiels

Il s'agit d'avoir un modèle formel permettant de raisonner sur les comportements d'un modèle UML. Pour l'instant, la thèse d'A. Le Guennec s'est concentrée sur l'obtention d'un codage sous forme de système de transitions (la génération d'un code de simulation séquentiel) à partir du diagramme de classes et des state-charts associés. BDL (formalisme pivot développé dans le cadre d'une collaboration entre les équipes Pampa et Ep-Atr de l'Irisa), lui, va plus loin, en proposant une sémantique de graphes avec un ensemble d'opérateurs de composition (avec une vision synchrone essentiellement). Notre objectif est ici intermédiaire : les actions des objets peuvent être vues comme des « flèches » d'un graphe (avec des pré et post conditions). Les règles d'assemblage de ces flèches dans le temps définissent la sémantique opérationnelle du modèle qui peut être d'ordre partiel. Différentes applications sont possibles. La première que j'étudierai est naturellement la synthèse des tests parallèles.

III.5.3. Diagnostic dans les systèmes répartis

Il s'agit de continuer à exploiter les idées de construction de trajectoires d'ordre partiel d'une spécification, guidée par l'observation partielle d'événements qui a été mis au point dans le cadre du projet Magda, au croisement des techniques de l'automatique, de l'IA et de l'informatique répartie. Mon apport consistera à participer à la « robustification » des algorithmes pour prendre en compte des modèles incomplètement spécifiés donnés par les référentiels de gestion de réseaux telecom, exprimés en UML. Une étude est en cours pour spécifier les pièces du puzzle de Viterbi en UML et générer automatiquement des règles activables depuis le logiciel Jrules de la société Ilog, logiciel dans lequel nous souhaitons embarquer notre algorithme de diagnostic.

III.6. Projets à plus long terme

Dans le cadre de la mobilité vers l'enseignement supérieur que j'ai entamée (ENS Cachan, antenne de Bretagne), et avec l'objectif de construire des projets communs renforçant les liens entre l'IRISA et l'ENS, je réfléchis à la possibilité de lancer un nouveau projet scientifique d'envergure. La fin réussie du projet PAMPA et mon départ de l'équipe de direction de l'IRISA me rendent en effet plus disponible. Le thème scientifique de l'invention de modèles et outils pour la supervision de grands systèmes répartis me semble particulièrement riche, ambitieux et porteur. Il s'agit de participer à la direction générale d'un contrôle des réseaux de plus en plus réparti. L'intérêt et les compétences acquises en matière de modèles d'ordres partiels et de diagnostic pourraient être au cœur d'un tel projet et mobiliser plusieurs autres chercheurs.

III.7. Références générales

- [1] Arnold, A.,
“ *Systèmes de transitions finis et sémantiques de processus communicants* ”
Masson, 1992, Etudes et recherches en informatique. 196 pages.
- [2] Jard, C.,
“ *Vérification dynamique des protocoles* ”
Habilitation à diriger les recherches de l'université de Rennes 1, 1994. Irisa coll. 156 pages.
- [3] Lamport, L.,
“ *Time, clocks and the ordering of events in a distributed system* ”
Communications of the ACM, (21):7, pp. 558-565, 1978.
- [4] Davey, B.A. and Priestley, H.A.,
“ *Introduction to Lattices and Order* ”
Cambridge University Press, 1990.
- [5] Tretmans, J.,
“ *Test Generation with Inputs, Outputs and Repetitive Quiescence* ”
Software - Concepts and Tools, Springer Verlag, Vol. 17, 1996.
- [6] Tarjan, R.,
“ *Depth-first search and linear graph algorithms* ”
SIAM Journal of Computing, (1):2, pp. 146-160, 1972.

IV. AUTRES ACTIVITES LIEES AU METIER DE CHERCHEUR

IV.1. Direction de recherches, encadrement de thèses

Je ne mentionne pas dans ce dossier, l'encadrement de stagiaires de DEA, DESS ou d'écoles d'ingénieurs. Les thèses citées se sont effectuées sous mon encadrement. Jusqu'en 1994 (année de mon habilitation), Michel Raynal a été le directeur de thèse officiel mais m'a confié l'entière responsabilité de l'encadrement scientifique. Depuis 1995, plusieurs thèses sont co-encadrées par des chercheurs de mon équipe.

- Omar Drissi-Kaitouni : Octobre 85 - Mai 88. Nu 207. Sujet : *La génération d'observateurs pour la validation des propriétés des systèmes distribués*. Thèse soutenue le 3 mai 1988, jury : L. Kott (président), P. Azéma, J.P. Banâtre (rapporteurs), Gv. Bochmann, R. Castanet, C. Jard et M. Raynal. O. Drissi est actuellement directeur des services informatiques d'une grande banque marocaine.
- Jean-François Monin : Octobre 84 - Novembre 88. Nu 292. Sujet : *Programmation en logique et compilation de protocoles : le simulateur Véda*. Thèse soutenue le 25 novembre 1988, jury : L. Kott (président), L.Trilling et J. Voiron (rapporteurs), P. Deransart, D. Herman, C. Jard et J.M. Pitié. J.F. Monin est actuellement ingénieur-chercheur à France Telecom R&D, expert sénior.
- Roland Groz : Octobre 84 - Janvier 89. Nu 194. Sujet : *Vérification de propriétés logiques des protocoles et systèmes répartis par observation de simulations*. Thèse soutenue le 20 janvier 1989, jury : D. Herman (président), P. Darondeau, P. Dembinski (rapporteurs), C. Jard, L.Kott, J.M. Pitié, B. Plateau et J.Sifakis. Après avoir été responsable de département à France Telecom R&D et responsable du pôle de recherche Génie Logiciel, R. Groz est actuellement professeur associé à l'ENSIMAG.
- Jean-Marc Pageot : Octobre 85 - Juin 89. Nu 342. Sujet : *Guider la simulation pour valider les protocoles*. Thèse soutenue le 29 juin 1989, jury : M. Raynal (président), M. Diaz, R. Marie (rapporteurs), J.M. Ayache, S. Budkowski, C. Jard et R. Groz. Thèse Cifre avec Bull. J.M. Pageot est actuellement ingénieur-chercheur à France Telecom R&D.
- Jean-Marc Jézéquel : Octobre 86 - Octobre 89. Nu 404. Sujet : *Outils pour l'expérimentation d'algorithmes distribués sur machines parallèles*. Thèse soutenue le 19 octobre 1989, jury : L. Kott (président), F. André, M. Cosnard (rapporteurs), A. Duda, C.Jard, G. Memmi et M. Raynal. J.M. Jézéquel a été recruté comme chargé de recherche au CNRS dans mon groupe en octobre 1990. Il est actuellement professeur de l'université de Rennes et responsable du projet (équipe) Triskell à l'IRISA.
- Thierry Jéron : Octobre 88 - Mai 91. Nu 630. Sujet : *Contribution à la validation des protocoles : test d'infinitude et vérification à la volée*. Thèse soutenue le 13 mai 1991, jury : J.P. Banâtre (président), A. Finkel, J.Sifakis, S. Graf (rapporteurs), B.Algayres, C. Jard, M. Raynal et P. Wolper. T. Jéron a été recruté comme chargé de recherche à l'INRIA dans mon groupe en octobre 1992. Il est maintenant responsable du projet (équipe) Vertecs à l'IRISA.
- Claire Diehl : Octobre 89 - Septembre 92. Nu 850. Sujet : *Analyse de la relation de causalité dans les exécutions réparties*. Thèse soutenue le 14 septembre 1992, jury : J.P. Banâtre (président), M. Habib, B. Plateau (rapporteurs), C.Jard, F. Mattern et M. Raynal. C. Diehl est actuellement chef de projet chez Philips.
- Benoît Caillaud : Octobre 89 - Juin 94. Nu 1126. Sujet : *Contribution à la modélisation du SPMD : la distribution asynchrone d'automates*. Thèse soutenue le 27 juin 1994, jury : J.P. Banâtre (président), L. Bougé, P. Caspi (rapporteurs), F. André, C. Jard, P. Le Guernic et M. Raynal. B. Caillaud a été recruté en 1996 comme chargé de recherche à l'INRIA dans mon projet. Il est maintenant responsable du projet (équipe) S4 à l'IRISA.
- Cyrille Bateau : Octobre 91 - Juillet 95. Nu 1395. Sujet : *Distribution automatique de programmes séquentiels : étude structurelle et expérimentale*. Thèse soutenue le 4 juillet 1995, jury : J.P. Banâtre (président), B. Plateau, O. Roux (rapporteurs), F. André, M. Ducassé, C. Jard et R. Thoraval. C. Bateau est actuellement ingénieur de recherche dans une société informatique française.
- Guy-Vincent Jourdan : Octobre 92 - Octobre 95. Nu 1361. Sujet : *L'analyse d'exécutions réparties en utilisant la théorie de l'ordre*. Thèse soutenue le 2 octobre 1995, jury : J.P. Banâtre (président), M. Habib, F. Mattern (rapporteurs), F. André, C. Jard, Y. Lepape, J.X. Rampon et M. Raynal. G.V. Jourdan est PDG d'une société informatique canadienne.
- Yves-Marie Quemener : Octobre 93 - Novembre 96. Nu 1681. Sujet : *Vérification de protocoles à espace d'états infini représentable par une grammaire de graphes*. Thèse soutenue le 22 novembre 1996, jury : J.P. Banâtre

- (président), A. Arnold, A. Bouajjani (rapporteurs), O. Burkart, C. Jard et T. Jérón. Co-encadrement avec T. Jérón. YM. Quemener est actuellement ingénieur de recherche à France Telecom R&D.
- Renée Boubour : Octobre 94 - Novembre 97. Nu 1827. Sujet : *Suivi de pannes par corrélation causale d'alarmes dans les systèmes répartis : application aux réseaux de télécommunication*. Thèse soutenue le 17 Octobre 1997, jury : J.P. Banâtre (président), A. Schaff, R. Valette (rapporteurs), A. Benveniste, C. Dousson, C. Jard. R. Boubour est actuellement chef de projet chez Dassault Data Systems.
 - Jean-Lin Pacherie : Décembre 1994 - Décembre 1997. Nu 1889. Sujet : *La génération de codes parallèles avec des techniques objets*. Thèse soutenue le 21 décembre 1997, jury : J.P. Banâtre (président), D. Trystram, R. Guerraoui (rapporteurs), JO. Coplien, C. Jard et JM. Jézéquel. Co-encadrement avec JM. Jézéquel. JL. Pacherie est actuellement ingénieur chez TMM.
 - Pierre Morel : Octobre 1996 – Février 2000. Nu 2320. Sujet : *Une algorithmique efficace pour la génération automatique de tests de conformité*. Thèse soutenue le 17 février 2000, jury : J.P. Banâtre (président), JC. Fernandez, R. Castanet (rapporteurs), A. Kerbrat, T. Jérón et C. Jard. Co-encadrement avec T. Jérón. P. Morel est actuellement ingénieur chez Dassault Data Systems.
 - Loïc Héliouët : Octobre 1997 - Octobre 2000. Sujet : *Analyse des exigences des systèmes répartis exprimées par des langages de scénarios*. Thèse soutenue le 16 octobre 2000, jury : M. Raynal (président), F. Khendec, A. Muscholl (rapporteurs), D. Vincent, B. Caillaud et C. Jard. Co-encadrement avec B. Caillaud. L. Héliouët a été recruté chargé de recherche à l'Inria dans l'équipe Triskell.
 - Alain Le Guennec : Octobre 1997 – Mars 2001. Sujet : *Génie logiciel et méthodes formelles avec UML. Spécification, validation et génération de tests*. Thèse soutenue le 29 juin 2001, jury : A. Benveniste (président), J. Bézivin, L. Laknech (rapporteurs), A. Kerbrat, JM. Jézéquel, C. Jard. Co-encadrement avec JM. Jézéquel. A. Le Guennec est actuellement ingénieur de recherche dans la société Telelogic.
 - Elena Zinovieva : Octobre 2000 - . Sujet : *Génération de tests symboliques*. Thèse en cours. Co-encadrement avec V. Rusu.
 - Valéry Tschaen : Octobre 2001 - . Sujet : *Test et contrôle*. Thèse en cours. Co-encadrement avec T. Jérón.

IV.2. Participation à des jurys de thèse

En dehors des jurys de thèse des étudiants que j'ai encadrés, j'ai été membre des jurys de thèse suivants, soit en qualité de rapporteur, soit en tant qu'examinateur, soit en tant que président.

En tant que rapporteur :

- René Thoraval : *Les principes et mécanismes fondamentaux de l'exclusion mutuelle équitable dans les systèmes distribués*. Mars 1988. Université de Rennes. IRISA.
- Laurent Steffan : *Conception et spécification de protocoles de communication pour applications temps réel*. Mai 1988. Université de Nantes. LAN.
- Rachida Dssouli : *L'observation dans les systèmes distribués*. Décembre 1988. PhD Université de Montréal. Canada. IRO.
- Jean-Louis Pizat : *Outils pour la programmation d'un multiprocesseurs à mémoire distribuée*. Février 1989. Université de Bordeaux. LABRI.
- Gilles Lesventes : *Systèmes d'automates à compteurs et semi-linéarité des ensembles d'états accessibles*. Juin 1989. Université de Rennes. IRISA.
- Christine Morin : *Protocole d'appel de multiprocédure à distance dans le système Gothic : définition et mise en œuvre*. Décembre 1990. Université de Rennes. IRISA.
- Mohamed Hajjam el Hassani : *Un protocole de transport multipoint fiable et efficace*. Février 1991. Université de Strasbourg.
- Pierre Moukeli : *Conception et mise en œuvre d'un système de mise au point intégré à l'environnement CNet pour les machines multi-processeurs SuperNode*. Juillet 1992. Université de Lyon, ENSL/LIP.
- Christian Antonescu : *Synthèse des spécifications de protocoles à partir des spécifications de service*. Mars 1993. PhD de l'université de Montréal, Canada. IRO.
- François Laroussinie : *Les logiques temporelles sur le passé*. Novembre 1994. Université de Grenoble. IMAG/LIFIA.
- Yves Arrouye : *Environnements de visualisation pour l'évaluation des performances des systèmes parallèles : étude, conception et réalisation*. Novembre 1995. Université de Grenoble. IMAG/LMC.

- Eric Mailet : *Le traçage logiciel d'applications parallèles : conception et ajustement de qualité*. Septembre 1996, Université de Grenoble. IMAG/LMC.
- Jan Parissis : *Le test de logiciels synchrones spécifiés en Lustre*. Septembre 1996. Université de Grenoble. IMAG/LSR.
- François Michel : *Symétries d'architecture et de données*. Validation de systèmes répartis. Décembre 1996. Université de Toulouse. LAAS/OLC.
- Akim Kahlouche : *Méthodologie de développement des protocoles de communication et des applications réparties. Vers une approche de synthèse*. Juillet 1997, Université de Saint Etienne.
- Luis Allan Kunstle : *Raisonnement temporel basé sur les réseaux de Petri pour des systèmes manipulant des ressources*. Octobre 1997. Université de Toulouse. LAAS.
- Barbara Heyd : *Application de la théorie des types et du démonstrateur COQ à la vérification de programmes parallèles*. Décembre 1997. Université de Nancy.
- Emanuel Melin : *Traitement de l'irrégularité dans la parallélisation de code séquentiel par distribution des données*. Février 1998. Université d'Orléans.
- Conrado Daws : *Méthodes d'analyse de systèmes temporisés : de la théorie à la pratique*. Octobre 1998, Université de Grenoble, Vérimag.
- Mihaela Sighireanu : *Contribution à la définition et à l'implémentation du langage "Extended Lotos"*. Janvier 1999, Université de Grenoble, Inria.
- Ahmed Bouajjani : *Vérification au delà du cadre des systèmes finis*. Janvier 1999, HDR, Université de Grenoble, Vérimag.
- Michel Bourdelles : *Analyse de systèmes réactifs synchrones à des fins de vérification : application au langage Esterel*. Mai 1999, Université de Nice, Inria.
- Solofo Ramangalahy : *Test de conformité pour des spécifications à base d'automates : une approche par la théorie des jeux*. Octobre 1999, Université d'Orléans.
- Saddek Bensalem : *Techniques algorithmiques et déductives pour la vérification des systèmes réactifs*. Mai 2000, HDR, Université de Grenoble, Vérimag.
- Nicolas Zuanon : *Test efficace de spécifications de services de télécommunication*. Juin 2000, Université de Grenoble, LSR.
- Athmane Touag : *Accélération de la génération de tests de protocoles par agrégation de méthodes hétérogènes*. Septembre 2000, Université Paris 6.
- Brian Nielsen : *Specification and Test of Real-Time Systems*. Octobre 2000, PhD Aalborg university.
- Roland Groz : *Hypothèses pyrthoniennes sur quelques aspects du génie logiciel en télécommunications*. Décembre 2000, HDR, Université de Bordeaux.
- Philippe Schnoebelen : *Spécification et vérification des systèmes concurrents*. Septembre 2001, HDR, Université Paris 7.
- Eric Thierry : *Sur quelques interactions entre structures de données et algorithmes efficaces pour les ordres et les graphes*. Octobre 2001, Université de Montpellier.
- Dominique Ambroise : *Marella ou la simulation guidée par les structures d'événements premières*. Octobre 2001, Université d'Orsay.
- Patricia Bouyer : *Modèles et algorithmes pour la vérification des systèmes temporisés*. Avril 2002, ENS Cachan.
- Constantin Ghirvu : *Génération automatique de tests de conformité pour les protocoles de communication*. Juillet 2002, Université de Grenoble.
- Fabien Peureux : *Génération de tests aux limites à partir de spécifications B en Programmation Logique avec Contraintes ensemblistes*. Décembre 2002, Université de Franche-Comté.

En tant qu'examineur :

- Christine Ecault : *Une expérience dans la spécification et la validation des systèmes distribués*. Avril 1985, Université de Rennes, IRISA.
- Arnaud Dupeux : *Spécification et implantation des protocoles OSI : une expérience de la couche transport dans le langage ADA*. Mars 1986. Université de Bordeaux. LABRI.
- Pierre Maviel : *Définition de classes d'environnements réseaux pour la simulation d'algorithmes distribués*. Février 1987. Université de Paris 6.
- Jean-Xavier Rampon : *Mesures de concurrence et extensions d'intervalles*. Février 1991. Université de Montpellier. LIRMM.

- Henry Thomas : *Une approche de la compilation de programmes séquentiels pour machines à mémoire distribuée*. Juin 1991. Université de Rennes. IRISA.
- Laurent Mounier : *Méthodes de vérification de spécifications comportementales : étude et mise en œuvre*. Janvier 1992. Université de Grenoble. VERIMAG.
- Fabrice Mourlin : *Conception et réalisation d'outils d'aide à la programmation sur réseaux de Transputers*. Novembre 1992. Université de Paris XI (Orsay). LRI.
- Didier Tranchier : *Fast Reservation Protocol/DT : multiplexage statistique dans les réseaux ATM*. Juin 1993. Université de Rennes. IRISA.
- Michel Hurfin : *L'analyse des calculs répartis : le cas de la reexécution*. Octobre 1993. Université de Rennes. IRISA.
- Céline Valot : *Le codage des ordres partiels : application aux exécutions réparties*. Octobre 1994. Université Pierre et Marie Curie. MASI.
- Alain Kerbrat : *Les méthodes symboliques pour la vérification*. Novembre 1994. Université de Grenoble. VERIMAG.
- Vlad Rusu : *Vérification temporelle de programmes Electre*. Janvier 1996. Université de Nantes. LAN.
- Lionel Van Aertrick : *Une méthode et un outil pour l'aide à la génération de jeux de tests de logiciels*. Janvier 1998. Université de Rennes. IRISA.
- Marius Bozga : *Vérification symbolique pour les protocoles de communication*. Décembre 1999, Université de Grenoble, Vérimag.
- Aurore Annichini-Collomb : *Vérification d'automates étendus : algorithmes d'analyse symbolique et mise en œuvre*. Décembre 2001, Université de Grenoble.
- César Viho : *Contribution au test des systèmes répartis*. Décembre 2002, Université de Rennes. HDR.

En tant que président :

- Laurent Tromp : *Surveillance et diagnostic de systèmes industriels complexes : une approche hybride numérique/symbolique*. Janvier 2000, Université de Rennes, IRISA.
- Yannick Pencolé : *Diagnostic décentralisé de systèmes à événements discrets : application aux réseaux de télécommunications*. Juin 2002, Université de Rennes, IRISA.

IV.3. Autres responsabilités scientifiques

Je consacre une partie significative de mon temps aux tâches d'animation-administration de la recherche, tant au niveau local qu'au niveau national :

- j'ai été membre élu du conseil de laboratoire de l'Irisa de 1990 à 1998.
- membre élu par ce conseil depuis 1990 pour participer au comité des projets du laboratoire. Depuis 1998, je suis nommé.
- j'ai été membre de l'équipe de direction de l'Irisa pendant 3 ans (1999-2002).
- J'ai été président de la commission personnel de l'Irisa, commission qui instruit scientifiquement les dossiers de recrutement des personnels scientifiques temporaires (doctorants, post-doctorants, invités, ingénieurs-experts, ingénieurs associés, délégations, détachements). Soit environ 150 dossiers par an. C'est à ce titre que j'ai fait partie de l'équipe de direction du laboratoire. J'ai été membre de la commission nationale des détachements à l'INRIA.
- j'ai été membre élu à la commission de spécialistes de l'université de Rennes (1987 - 1995), membre nommé dans celle de l'université de Brest (1992 - 1995), ainsi qu'à celle d'Orsay (1993 - 1995). De 1998 à 2002, membre élu de nouveau à la commission de l'université de Rennes et nommé à la commission de l'INSA de Rennes.
- j'ai été élu au Comité National de la Recherche Scientifique, en tant que CR de 1991 à 1995 (section 07). En tant que membre de ce comité, j'ai participé à quelques comités scientifiques de formations associées au CNRS, et au comité du programme inter-disciplinaire sur la Ville. Je suis resté membre du comité de pilotage de l'action sur les nouvelles technologies et la sécurité dans les villes, soutenue par l'IHESI (Institut des hautes études sur la sécurité intérieure) et le CNRS jusqu'à la fin du programme (1998). Depuis 2000, je suis élu de nouveau, en tant que DR, au Comité National de la Recherche Scientifique (section 07). Je suis membre du bureau de la section.
- je suis élu au conseil scientifique du nouveau département STIC (Science et Technologie de l'Information et de la Communication) du CNRS depuis septembre 2001.
- j'ai été un membre actif du PRC/GDR C3 depuis sa création. J'y ai lancé et animé plusieurs actions de coopération : dans l'ordre chronologique, le projet *Atlas d'algorithmes distribués*, l'action sur la machine TNode

et enfin l'action sur l'analyse de traces d'exécutions (qui fut transformé en un projet inter-PRC). Sur chacune de ces opérations, une dizaine d'équipes nationales y participaient. A la suite, dans le GDR-PRC PRS, j'ai été responsable d'un projet sur la vérification et le test des systèmes répartis ; celui-ci était aussi reconnu par le GDR-PRC AMI. Jusqu'en 2002, j'ai co-animé (avec MC. Gaudel et P. Thévenod) un groupe de travail national sur le test et objets dans le cadre du GDR-PRC ALP.

- De 1995 à 2001, et c'est la tâche prioritaire d'animation qui m'a occupé, j'ai pris la direction de l'équipe (projet) INRIA/CNRS/Enseignement Supérieur PAMPA (*modèles et outils pour la programmation des architectures parallèles réparties*). Ce projet avait pour objectif de contribuer à la maîtrise du développement des logiciels répartis sur les réseaux de processeurs. Les activités de modélisation des phénomènes répartis et d'analyse (statique ou dynamique) des comportements engendrés que j'ai contribué à développer y occupaient une place centrale. Elles ont été largement étendues par les travaux d'autres chercheurs de l'équipe, notamment sur les aspects modélisation à objets. PAMPA a rassemblé en 2000 l'activité de plus de 25 personnes, dont 1 professeur (JM. Jézéquel), 3 Maîtres de Conférences (C. Viho, Y. Le Traon, N. Plouzeau), 3 chercheurs INRIA (T. Jéron, B. Caillaud, V. Rusu), une dizaine de doctorants, quelques post-doctorants et 6 ingénieurs en CDD. En juillet 2001, ce projet s'est arrêté pour contribuer fortement à la naissance de nouvelles équipes en génie logiciel à l'Irisa, autour de JM. Jézéquel, T. Jéron et B. Caillaud. J'effectue actuellement ma recherche dans la nouvelle équipe Triskell, dirigée par JM. Jézéquel, dont le sujet est la maîtrise de l'assemblage fiable de composants logiciels, principalement dans le domaine des télécommunications.

IV.3. Valorisation de la recherche

IV.3.1. Contrats industriels

Mon activité de recherche est de nature modélisation et algorithmique ; elle vise à concevoir des méthodes destinées dans le meilleur des cas à être intégrées quelques années plus tard dans des environnements de validation pour laboratoires industriels. C'est pourquoi, au delà de publications souvent à cheval entre le milieu de l'informatique théorique et celui des systèmes répartis et des réseaux, j'accorde une importance particulière à l'approfondissement de logiciels prototypes et au transfert de savoir-faire vers le milieu industriel. Cela passe aussi par l'étude de vraies applications.

- au CNET, j'ai lancé et suivi le contrat de réalisation de l'outil de vérification *Xésar*, commandé à l'IMAG/LGI et à CAP-Sogeti-Innovation. Il s'agissait de faire passer cet outil pionnier d'un stade de démonstration utilisant un langage jouet au contexte réel de la vérification de programmes *Estelle* (1982 - 1986).
- depuis 1987, à l'IRISA, j'ai été associé à l'industrialisation du logiciel *Véda*, avec la société Vérilog, sous contrat du ministère de la recherche et de la technologie. Le fait intéressant est que Vérilog commercialise maintenant avec *Véda2* puis *ObjectGéode*, des environnements de spécification et vérification de protocoles intégrant plusieurs des algorithmes et concepts que nous avons conçus (les observateurs, la simulation intensive, la vérification à-la-volée, la représentation des files). Cette coopération fructueuse avec Vérilog, Telelogic maintenant, a continué sur le sujet de l'utilisation de l'algorithmique de la vérification pour la génération automatique de tests de conformité pour les protocoles : notre prototype *TGV* existe sous le nom de *TestComposer* au catalogue de Telelogic. Ce transfert a été en partie financé par les services techniques de la direction générale de l'armement et France Telecom.
- j'ai été l'instigateur et le responsable de plusieurs contrats de conseil dans le domaine de la validation de protocoles. A ces contrats était associée une mise à disposition de logiciels prototypes développés dans mon groupe (notamment *Echidna* et *TGV*) pour évaluation industrielle. Les industriels engagés furent :
 - *Transpac* : nous avons participé à la validation de nouveaux protocoles sur le réseau (1988 - 1990).
 - le *Celar* : c'est le centre de recherche de l'armement. Je conseille depuis plusieurs années leurs activités en matière de validation de protocoles. Le *Celar* agissant en tant que donneur d'ordres, ceci me donne l'occasion de rencontrer de nombreux industriels du domaine. Il a soutenu fortement notre recherche en matière d'observation des comportements répartis dans les réseaux, avec l'objectif d'aider la recherche fondamentale sur le thème de l'interopérabilité (1990 - 1995). Il finance actuellement une thèse sur le sujet (1998-2001).
 - *Matra-Marconi-Space* à Toulouse. Nous avons effectué une évaluation conjointe de *Echidna* sur l'expérimentation de protocoles dans le domaine du spatial (1989).
 - le *CNET* à Lannion, soutient depuis 1994 nos études sur l'application de l'algorithmique répartie au problème de diagnostic dans les réseaux (1994 - 1998), relayé ensuite par le projet RNTL Magda.

- *Alcatel* : nous sommes engagés depuis 1995 dans une action de partenariat INRIA-ALCATEL sur le thème de la modélisation et vérification d'applications de télécommunications au dessus d'une plateforme offrant une interface de type CORBA. Je pilote pour l'INRIA cette action nationale, appelée REUTEL. REUTEL a fait participer 5 équipes et a permis de financer plusieurs doctorants et ingénieurs.
- *GIE BULL-INRIA Dyade* : je suis co-responsable depuis 1995 de l'action VASY (puis FormalFame) sur l'utilisation de méthodes formelles (LOTOS) pour vérifier et tester des architectures multi-processeurs en cours de développement chez Bull.
- en complément de ces partenariats long-terme avec l'industrie, je voudrais citer ma participation et responsabilité dans la conduite des projets RNRT Magda (*Supervision de réseaux*) (1998-2003), RNTL COTE (*Composants et Tests*) (2000-2002) et CEE Esprit Modistarc (*Test de protocoles embarqués dans l'automobile*) (1997-2000).

Depuis 1986, l'ensemble du financement industriel sur ces contrats pour l'IRISA a représenté environ 500 kF par an avec une augmentation nette sur la période 1998-2001, liée à l'augmentation globale de l'activité de mon équipe, et aussi au fait que certains contrats concernaient plusieurs équipes.

IV.3.2. Contrats institutionnels et coopération internationale

- j'ai été responsable scientifique pour plusieurs actions de soutien institutionnel : dans le PRC C³, le pôle réseau de la DRED.
- en 1994 et 1995, j'ai participé à la direction d'un projet national inter-PRC, appelé "Trace" (J. Chassin de l'IMAG/LMC à Grenoble en fut le responsable), réunissant une trentaine de personnes. Ce projet a développé un environnement d'analyse de traces d'exécution parallèles, dans lesquels nous avons démontré nos recherches sur l'analyse des dépendances causales.
- j'ai défini, en coopération avec B. Plateau à Grenoble et D. Zernik au Technion à Haifa, une action de coopération Franco-Israélienne, qui a tenu de 1992 à 1994; de même, en liaison avec R. Groz du CNET et G. Bochmann de l'université de Montréal, une action de coopération Franco-Québécoise. Ces actions ont permis des échanges de chercheurs entre nos laboratoires. C'est le cas aussi actuellement avec l'université de Twente (E. Brinskma) et le SRI Stanford (J. Rushby).
- de 1992 à 1996, un contrat CEE/HCM m'a permis d'accueillir 3 chercheurs post-doctorants, successivement sur les thèmes de la modélisation, l'évaluation et la vérification des systèmes répartis.
- j'ai participé au programme européen COST 247 sur les méthodes formelles de 1993 à 1997.
- j'ai été co-responsable (avec J. Sifakis) d'un projet national CNRS-DRET-MENRT (FORMA) sur l'utilisation de méthodes formelles pour le développement de systèmes critiques, en collaboration avec plusieurs laboratoires académiques et industriels (1997-2000).
- J'ai été responsable d'un projet sur le test dans le cadre du programme Télécom du CNRS (1999-2001). Je suis aussi le responsable Irisa d'une action spécifique du département STIC sur le thème du test de robustesse (2001-2002).
- j'ai participé à une action incitative Inria (ARC FISC, 2001-2002) sur l'étude formelle des langages de scénarios.

IV.3.3. Activités de conseil et d'expertise

Après mon départ du CNET, j'ai continué à suivre pendant un an les activités du département Evaluation et Validation de Protocoles et l'industrialisation de Véda en tant que chargé d'études du CNET de Lannion.

J'ai assuré en 1989 à titre personnel une activité de conseil pour le compte de la société Bull sur le thème de la validation de réseaux locaux temps-réels, dans le cadre d'une prestation de service à destination de la société April, située à Grenoble.

Depuis 2000, j'interviens ponctuellement pour le compte de la société Synesis dans le cadre d'expertise de haut-niveau sur les méthodes de développement auprès de grandes sociétés. Je suis actuellement membre du conseil stratégique de la jeune société TNI-Valiosys sur le thème de la vérification formelle.

J'ai aussi participé à quelques expertises dans le cadre :

- du programme européen Esprit (BRA 93, LTR 95-96),

- de programmes nationaux des ministères (RNRT, RNTL),
- du NSERC (Canada) et NSF (USA),
- du programme de réseaux d'excellence (GISST) du gouvernement Canadien.

Récemment, je suis devenu membre du comité scientifique du centre de recherche en informatique de Montréal. Je suis membre du comité de pilotage du RTP CNRS/STIC sur les systèmes embarqués où je représente l'Irisa.

IV.3.4. Valorisation de logiciel

Trois réalisations ont été menées jusqu'à un stade avancé de prototype :

- La première est le logiciel de simulation Véda, fondé sur le langage normalisé Estelle pour la description formelle des protocoles, dont la première version construite avec JF. Monin et R. Groz était achevée en 1985. La licence a été transférée à la société Vérilog. Le produit a ensuite continué à évoluer en partenariat avec l'IRISA, le CNET et soutien de l'état. Jusqu'en 1995, Véda figurait au catalogue de la société. Actuellement, l'essentiel de la "technologie Véda" a été transférée dans l'outil Géode de manipulation de programmes SDL, vendu assez largement par Telelogic maintenant. Auparavant, dans la version CNET, Véda avait déjà été bien diffusé dans le milieu académique en France (notamment dans le cadre du pôle C³ sur les algorithmes répartis) et à l'étranger (plus de 60 sites).
- Le deuxième outil est le logiciel Echidna, qui implante automatiquement des applications réparties écrites en Estelle, sur différents réseaux de processeurs. Achevé en 1989 avec la collaboration de JM. Jézéquel, il a été diffusé à plus de trente exemplaires dans le milieu académique. Il a fait l'objet d'évaluations industrielles dans le cadre de licences d'utilisation cédées à Transpac et Matra-Espace. Le transfert industriel ne s'est pas produit. Curieusement, c'est l'application à l'enseignement qui s'est le plus développé, et qui vit encore malgré l'obsolescence du langage Estelle. A noter aussi qu'une partie importante de l'exécutif associé à Echidna a été récupéré pour construire la POM (Parallel Observable Machine), un support d'exécution parallèle transparent comparable à PVM.
- Le troisième prototype notable est TGV (Test Generation using Verification), un outil capable, à partir d'une spécification SDL (ou Lotos, et maintenant UML), et d'objectifs de test formalisés par des automates, de produire une suite de tests TTCN (TTCN est une notation normalisée pour décrire des tests de conformité de protocoles) testant la conformité d'une implantation de protocole par rapport à sa spécification formelle. Réalisé rapidement dans le cadre d'une coopération Cap-Sesa, Vérilog, INRIA, CNET et le service technique de l'armée, en 1996, il a progressivement mûri jusqu'à pouvoir être transféré dans l'industrie. Une partie de TGV a été réalisée en collaboration avec le laboratoire Vérimag. TGV existe maintenant au catalogue de Telelogic sous le nom de TestComposer. Je prépare actuellement un nouveau prototype, appelé TGP, intégrant des fonctionnalités de génération de tests parallèles.

IV.3.5. Diffusion de l'information scientifique et technique

Organisation de colloques

- Ecole de printemps Inria/CNRS sur le thème de l'algorithmique parallèle et distribuée, La Colle/Loup Avril 1987, présidée par F. Baccelli, M. Raynal et moi-même. Organisation de plusieurs journées C³ également.
- Ecole d'informatique sur "calculs répartis et causalité", Roscoff, septembre 1993. 45 chercheurs. Organisée par B. Charron et moi-même, avec le soutien du CNRS.
- 4^{ème} colloque francophone sur l'ingénierie des protocoles. Rennes, mai 1995. 160 chercheurs. Organisé par P. Rolin et moi-même, avec le soutien du CNRS.
- MOVEP'96, 98, 2000 : Ecole sur la modélisation et la vérification des processus parallèles. Nantes, juin 1996. Organisée par O. Roux, B. Rozoy et moi-même. Nous avons recommencé cette manifestation en 1998. Devant le succès (120 participants et des orateurs de très haut niveau), l'édition 2000 est devenue une école européenne internationale. Les cours ont fait l'objet de l'édition d'un livre publié en 2001 dans la série LNCS Tutorial. L'édition 2002 s'est déroulée dans les mêmes conditions.
- EuroPar'96 : la conférence sur le parallélisme en Europe. J'ai été "local chair" du workshop sur "Distributed systems and algorithms". Lyon, Août 1996.
- 5^{ème} rencontres Jacques Cartier sur les méthodes formelles et leur utilisation sur les systèmes critiques. Montréal, octobre 1996. Co-présidées par G. Bochmann, M. Barbeau, M. Freedman, J. Sifakis et moi-même.

- Présidence du “Industrial Tutorials on Testing” sur le thème du test dans le cadre du congrès mondial sur les méthodes formelles (FM99) à Toulouse.
- Co-président (avec S. Graf) de la conférence SAM’2000 sur SDL et MSC, Grenoble.
- Président des doctoriales de Bretagne, prévues en Novembre 2003.

Comités de rédaction

J’ai été membre du comité de rédaction de deux revues nationales : *la lettre du Transputer et des calculateurs distribués* (devenue maintenant la revue *Calculateurs Parallèles*), publiée chez Hermès, et la revue *réseaux et informatique répartie*, publiée également chez Hermès jusqu’en 1995.

Je suis membre du comité scientifique de la revue “Annals of Telecommunications”.

J’ai été co-éditeur (avec H. Garavel et R. Groz) de deux numéros spéciaux de TSI sur l’application des méthodes formelles en 1997 et 1998. En 2000, j’ai été membre du comité de rédaction d’un numéro spécial de TSI sur le test.

Comités de programme

J’ai été membre des comités de programme suivants :

- 5^{ème} IFIP Workshop on Protocol Specification, Testing and Verification, Toulouse 1985 ; PSTV VI (Symposium qui a pris la suite du workshop précédent). Montréal 1986 ; PSTV X. Ottawa 1990.
- CFIP, Colloque Francophone sur l’Ingénierie des Protocoles. Bordeaux 1988 ; Pau 1991 ; Montréal 1993 ; Rabat 1996 ; Liège 1997 ; Nancy 1999 ; Toulouse 2000 ; Montréal 2002.
- Transputer. Besançon 1992 ; Besançon 1994.
- Renpar5. Brest 1993 ; Renpar6. Lyon 1994 ; Renpar7. Mons 1995 ; Renpar8. Bordeaux 1996 ; Renpar11. Rennes 1999 ; Renpar12 Besançon 2000.
- AAdebug, Automated Algorithmic Debugging. Mont-Saint-Michel 1995, Linköping 1997.
- GRES, Gestion de réseaux. Rennes 1997 ; Montréal 2000.
- AFADL, Assistance au développement formel de logiciels. Poitiers 1998 ; Grenoble 2000 ; Nancy 2001 ; Rennes 2003.
- SDL-Forum, Montréal 1999.
- MSR, Modélisation des systèmes réactifs. Membre du comité de pilotage de la conférence. Cachan 1998 ; Toulouse 2001 ; Metz 2003.
- TACAS, Tools and algorithms for the construction of systems (ETAPS). Berlin 2000 ; Genova 2001 ; Grenoble 2002.
- Testcom, Testing of communicating systems. Ottawa 2000 ; Berlin 2001.
- FORTE, Formal description techniques. Houston 2002 ; Berlin 2003.

Relectures

J’assure régulièrement la lecture d’articles pour des revues (ACM Toplas, ACM Tosem, Parallel Processing Letters, Information Processing Letters, IEEE Computer (lecteur attitré), TSI, TCS, Distributed Computing, ...) et de communications pour des colloques (1 par mois en moyenne). Je suis un des relecteurs pour les livres soumis pour édition à l’IEEE Computer Society Press.

Normalisation

J’ai participé de façon irrégulière aux comités de normalisation qui avaient en charge la définition du langage Estelle au CCITT d’une part et à l’ISO d’autre part. Outre des réunions aux quatre coins de la planète, il a fallu rédiger plusieurs contributions. Actuellement mon engagement est léger et se situe dans le cadre de l’OMG sur une proposition d’un profil “test” pour UML.

V. DOSSIER DE PUBLICATIONS

V.1 Ouvrages édités et chapitres de livres

[OE13] F. Cassez, C. Jard, F. Laroussinie and M. Ryan (eds).
Modeling and Verification of Parallel Processes, revised Tutorial Lectures, *LNCS Tutorials*, Springer, to appear in 2003.

[OE12] F. Cassez, C. Jard, F. Laroussinie and M. Ryan (eds).
Modeling and Verification of Parallel Processes, MOVEP'2002, *European Summer School*, Nantes, Jun. 2002.

[OE11] F. Cassez, C. Jard, B. Rozoy and M. Ryan (eds).
Modeling and Verification of Parallel Processes, revised Tutorial Lectures, *LNCS Tutorials*, Nu. 2067, Springer, Apr. 2001.

[OE10] S. Graf and C. Jard (eds).
SAM2000, 2nd Workshop on SDL and MSC, Grenoble, Jun. 2000.

[OE9] F. Cassez, C. Jard, B. Rozoy et M. Ryan (eds).
Verification and Modeling of Parallel Processes, MOVEP'2k, *European Summer School*, Nantes, Jun. 2000.

[OE8] F. Cassez, C. Jard, O. Roux et B. Rozoy (ed).
Modélisation et vérification des processus parallèles, *MOVEP'98, Actes de l'ecole d'été, ECN*, Juillet 1998.

[OE7] X. Blondel et C. Jard,
Application des techniques formelles au logiciel -- LDS appliqué à la modélisation et à la validation des processus communicants, *ARAGO-20, Observatoire français des techniques avancées*, Juin 1997.

[OE6] H. Garavel, R. Groz et C. Jard (ed).
Technique et science informatiques -- Méthodes formelles, validation de systèmes complexes,
Vol. 16, Juin 1997.

[OE5] Gv. Bochmann, C. Jard and J. Sifakis (ed).
Application of Formal Methods to System Development - Telecommunications, VLSI and Real-Time Computerized Control Systems, Montreal, Oct. 1996. Entretiens Jacques Cartier.

[OE4] C. Jard et P. Rolin (ed).
Colloque Francophone sur l'Ingénierie des Protocoles, Rennes, Mai 1995, Hermès.

[OE3] R. Keunings, F.-X. Roux, P. Leca, C. Jard (Eds.),
Applications Industrielles du Parallélisme, *Calculateurs Parallèles*, Hermes, Special Issue (1995)

[OE2] B. Charron-Bost et C. Jard (ed).
Ecole d'informatique sur l'algorithmique répartie -- calculs répartis et causalité, Roscoff, Sep. 1993.

[OE1] F. André et C. Jard (ed).
Datation et contrôle des exécutions réparties, *Bigre Nu 78*, Mars 1992. Colloque C3.

V.2. Publications dans des revues internationales

[RI20] A. Benveniste, E. Fabre, S. Haar and C. Jard.
Diagnosis of asynchronous discrete event systems, a net unfolding approach. *IEEE transactions on Automatic Control*, to appear in 2003.

- [RI19] C. Jard.
Synthesis of distributed testers from true-concurrency models of reactive systems. *International Journal of Information and Software Technology*, R. Dssouli, F. Khendec ed., Elsevier, 2002.
- [RI18] L. Hélouët, C. Jard and B. Caillaud.
An Event Structure Semantics for Message Sequence Charts. *Mathematical Structures in Computer Science (MSCS) journal*, vol. 12, pp. 377-403. Cambridge University Press. 2002.
- [RI17] C. Jard and T. Jérón.
An Educational Case Study in Protocol Verification and Distributed Observation. *Journal of Computer Science Education, ECASP Special Issue*, 10(3), 2000.
- [RI16] M. Bozga, JC. Fernandez, L. Ghirvu, C. Jard, T. Jérón, A. Kerbrat, P. Morel and L. Mounier.
Verification and Test Generation for the SSCOP Protocol. *Journal of Science of Computer Programming, Special Issue on The Application of Formal Methods in Industrial Critical Systems*, Vol. 36, pp. 27-52, 2000.
- [RI15] E. Fabre, A. Aghasaryan, A. Benveniste, R. Boubour and C. Jard.
Fault Detection and Diagnosis in Distributed Systems : an Approach by Partially Stochastic Petri Nets. *Journal of Discrete Events Dynamic Systems*, Vol. 8, pp. 203-231, 1998.
- [RI14] B. Caillaud, P. Caspi, A. Girault and C. Jard.
Distributing Automata for Asynchronous Networks of Processors. *European Journal on Automated Systems (JESA)*, Vol. 31, Nu. 3, pp. 503-524, 1997.
- [RI13] JC. Fernandez, C. Jard, T. Jérón and C. Viho.
An Experiment in Automatic Generation of Test Suites for Protocols with Verification Technology. *Science of Computer Programming, special issue on industrially relevant applications of formal analysis techniques*, 1997.
- [RI12] C. Jard and GV. Jourdan.
Incremental Transitive Dependency Tracking in Distributed Computations. *Parallel Processing Letters*, Vol. 6, Nu. 2, pp. 427-435, 1996.
- [RI11] C. Jard and JM. Jézéquel.
Building a Global Clock for Observing Computations in Distributed Memory Parallel Computers, *Concurrency Practice and Experience*, Vol. 1, Nu. 8, pp. 71-89, Jan. 1996.
- [RI10] E. Fromentin, C. Jard, GV. Jourdan and M. Raynal.
On-the-fly Analysis of Distributed Computations, *Information Processing Letters*, Nu. 54, pp. 267-274, 1995.
- [RI9] C. Jard, GV. Jourdan and JX. Rampon.
On-line Computations of the Ideal Lattice of Posets, *RAIRO ITA, Theoretical Informatics and Applications*, Vol. 29, Nu. 3, pp. 227-244, 1995.
- [RI8] GV. Jourdan, JX. Rampon and C. Jard.
Computing On-line the Lattice of Maximal Antichains of Posets, *Order*, Nu. 11, pp. 197-210, December 1994.
- [RI7] T. Jérón and C. Jard.
Testing for Unboundedness of Fifo Channels, *Theoretical Computer Science*, Vol 113, pp. 93-117, June 1993.
- [RI6] JC. Fernandez, C. Jard, T. Jérón and L. Mounier.
On-the-fly Verification of Finite Transition Systems, *Formal Methods in System Design*, Vol 1, pp. 251-273, 1993.
- [RI5] C. Jard and JM. Jézéquel.
Echidna, an Estelle-Compiler to Prototype Protocols on Distributed Computers, *Concurrency Practice and Experience*, Vol. 4, Nu, 5, pp. 377-397, Aug. 1992.

[RI4] C. Jard, JF. Monin and R. Groz.

Development of VEDA- a Prototyping Tool for Distributed Algorithms, *IEEE trans. on Software Engineering*, Vol. 14, Nu. 3, Mar. 1988.

[RI3] R. Groz, C. Jard, and C. Lassudrie.

Attacking a Complex Distributed Algorithm from Different Sides – an Experience with Complementary Validation Tools. *Computer Networks*, Vol. 10, Nu. 5, pp. 245-257, Dec. 1985.

[RI2] C. Jard and Gv. Bochmann.

An approach to Testing Specification. *The Journal of System and Software*, Nu. 3, pp. 315-323, 1983.

[RI1] Gv. Bochmann, E. Cerny, C. Jard, B. Sarikaya.

Experience with Formal Specifications using an Extended State Transition Model. *IEEE trans. on Communications*, Vol. 30, Nu. 12, pp. 2506-2513, 1982.

V.3. Publications dans des revues d'audience plus limitée

[RN7] C. Jard, JM. Jézéquel, A. Le Guennec and B. Caillaud.

Protocol Engineering using UML. *Annals of Telecom*, Vol. 54, 11-12, pp. 526-538, Nov. 1999

[RN6] T. Jéron, C. Jard, C. Viho, B. Caillaud, H. Kahlouche, P. Morel, JC. Fernandez, A. Kerbrat et M. Bozga.

Génération automatique de tests pour les protocoles : l'exemple de l'approche formelle de TGV. *Revue Electronique et Electricité*, 1999.

[RN5] C. Jard, JM. Jézéquel et F. Pennaneac'h.

Vers l'utilisation d'outils de validation de protocoles dans UML. *Technique et Science Informatiques*, Hermes, Vol. 17, Nu. 9, 1998.

[RN4] C. Jard.

Observation d'exécutions réparties, détection de propriétés d'ordonnancement. *Technique et Science Informatiques*, Hermes, Vol. 16, Nu. 2, p. 225-242, 1997.

[RN3] C. Diehl et C. Jard.

L'observation des exécutions réparties. Datation par intervalles. *Revue Réseaux et Informatique Répartie*, Hermès, Vol. 1, Nu. 4, 1992.

[RN2] JP. Courtiat, P. Dembinski, R. Groz et C. Jard.

Estelle, un langage ISO pour les algorithmes distribués et les protocoles. *Technique et Science Informatique*, Vol. 6, Nu. 2, 1987.

Version anglaise : *Technology and Science of Informatics*, Vol. 6, Nu. 5, pp. 311-324.

[RN1] G. Hébuterne et C. Jard.

Modélisation et analyse des algorithmes distribués et protocoles. *L'Echo des Recherches*, Nu 122, Dec. 1985.

V.4. Publications dans des colloques internationaux

[CI50] S. Pickin, C. Jard, Y. Le Traon, T. Jéron, JM. Jézéquel, A. Le Guennec.

System test synthesis from UML models of distributed software. *FORTE'2002, IFIP Int. Conf. on Formal description techniques*. Nov. 2002, Houston, Texas.

[CI49] A. Benveniste, E. Fabre, C. Jard, S. Haar.

Diagnosis of asynchronous discrete event systems, a net unfolding approach. *WODES'02: 6th Int. Work. on Discrete Event Systems*. Zaragoza, Spain. October 2-4 2002.

[CI48] Magda consortium.

XVIII World Telecommunications Congress, 22-27 September 2002 - Paris, France

[CI47] C. Jard and T. Jéron.

TGV: Theory, Principles and Algorithms.

The sixth world conference on integrated design and process technology. Pasadena, California, June 23-28, 2002.

[CI46] E. Fabre, A. Benveniste and C. Jard.

Distributed Diagnosis for Large Discrete Event Dynamic Systems.

15th IFAC World Congress on Automatic Control, Barcelona, July 21-26, 2002.

[CI45] C. Jard.

Principles of Distributed Test Synthesis based on True-concurrency Models. *Testcom'2002*. Berlin, Mar. 2002.

[CI44] S. Pickin, C. Jard, T. Heuillard, JM. Jézéquel and P. Desfray.

A UML-integrated Test Description Language for Component Testing. *Practical UML-based Rigorous Development Methods*, A. Moreira (editor), Totonto, Oct. 2001.

[CI43] C. Jard.

How to simulate HMSC. *SDL'01 : meeting UML, 10th SDL Forum*, LNCS, Copenhagen, Jun. 2001.

[CI42] E. Fabre, A. Benveniste, C. Jard, L. Ricker and M. Smith.

Distributed State Reconstruction for Discrete Event Systems. *39th IEEE Conf. on Detection and Control (CDC2000)*, Dec. 2000.

[CI41] C. Jard, T. Jéron and P. Morel.

Verification of Test Suites. *IFIP TC6/WG6.1 13th International Conference on Testing of Communicating Systems, TestCom 2000*, Ottawa, pp. 3-18, Aug. 2000.

[CI40] L. Tanguy, C. Viho and C. Jard.

Synthesizing Coordination Procedures for Distributed Testing of Distributed Systems. *ICDCS Workshop Distributed System Validation and Verification (DSVV'2000)*, TH. Lai (ed), IEEE Computer Society, pp. E67-E74, Taipei, Taiwan, ROC, Apr. 2000.

[CI39] L. Hélouët and C. Jard.

Conditions for Synthesis of Communicating Automata from HMSCs. *5th Int. Workshop on Formal Methods for Industrial Critical Systems (FMICS)*, ARE. Stefania-Gnesi, I. Schieferdecker (ed), GMD FOKUS, Apr. 2000.

[CI38] L. Hélouët, C. Jard and B. Caillaud.

An Effective Equivalence for Sets of Scenarios. *Proceedings of GRATRA'2000 (joint appligraph and getgrats workshop on graph transformation systems)*, ETAPS 2000, GT. H. Ehrig (ed), Berlin, Germany, Mar. 2000.

[CI37] C. Jard, T. Jéron, L. Tanguy and C. Viho.

Remote Testing can be as Powerful as Local Testing. *Methods for protocol engineering and distributed systems, FORTE XII/PSTV XIX'99*, Beijing, China, J. Wu, S. Chanson and Q. Gao (ed), Kluwer Academic Publishers, pp. 25-40, Oct. 1999.

[CI36] C. Jard and T. Jéron.

Verification and Distributed Observation of the Alternating Bit Protocol. *IFIP TC6/WG6.1 Joint International Conference on formal description techniques (Forte XI), and protocol specification, testing and verification (Pstv XVIII), special session on educational case studies in protocols*, Paris, 1998.

[CI35] C. Jard, T. Jéron, H. Kahlouche and C. Viho.

Towards Automatic Distribution of Testers for Distributed Conformance Testing. *IFIP TC6/WG6.1 Joint International Conference on formal description techniques (Forte XI), and protocol specification, testing and verification (Pstv XVIII)*, Paris, 1998.

- [CI34] A. Aghasaryan, E. Fabre, A. Benveniste, R. Boubour and C. Jard.
A “hybrid” Stochastic Petri Net Approach to Fault Diagnosis in Large Distributed Systems. *Mathematical Theory of Networks and Systems, MNTS'98*, Jul. 1998, Padova, Italy.
- [CI33] A. Benveniste, S. Gaubert and C. Jard.
Monotone Rational Series and Max-plus Algebraic Models of Real-time Systems. *4th Workshop on Discrete Event Systems, Wodes'98*, Cagliari, 1998.
- [CI32] A. Benveniste, C. Jard and S. Gaubert.
Algebraic Techniques for Timed Systems. *9th International Conference on Concurrency Theory, Concur'98*, Nice, 1998.
- [CI31] JP. Talpin, A. Benveniste, B. Caillaud, C. Jard, Z. Bouziane and H. Canon.
BDL, a Language of Distributed Reactive Objects. *Proceedings of the ISORC'98, The 1st IEEE International Symposium on Object-oriented Real-time Distributed Computing*, Kyoto, 1998.
- [CI30] R. Boubour, C. Jard, E. Fabre, A. Aghasaryan and A. Benveniste.
A Petri Net Approach to Fault Detection and Diagnosis in Distributed Systems. Part I, Application to Telecommunication Networks, Motivations, and Modelling. *Proceedings of the annual IEEE Control and Decision Conference (CDC'97)*, San-Diego, Oct. 1997.
- [CI29] E. Fabre, A. Aghasaryan, A. Benveniste, R. Boubour and C. Jard.
A Petri Net Approach to Fault Detection and Diagnosis in Distributed Systems. Part II, Extending Viterbi Algorithm and HMM Techniques to Petri Nets. *Proceedings of the annual IEEE Control and Decision Conference (CDC'97)*, San-Diego, Oct. 1997.
- [CI28] R. Boubour and C. Jard.
Fault Detection in Telecommunication Networks based on Petri Net Representation of Alarm Propagation. *Proceedings of the 18th Int. Conf. on Application and Theory of Petri Nets*, Toulouse, Jun. 1997.
- [CI27] JC. Fernandez, C. Jard, T. Jérón and C. Viho.
Using on-the-fly Verification Techniques for the Generation of Test Suites. *Computer Aided Verification (CAV96)*, Boston, Aug. 1996. LNCS 1102.
- [CI26] C. Bareau and C. Jard.
Dynamic Analysis of Parallelized Sequential Programs using Instrumented Semantics. *Parallel Environments for High Performance Computing, European School of Computer Science*, Grenoble, Apr. 1996.
- [CI25] C. Jard, JM. Jézéquel and L. Nédelka.
An Approach to Integrate Formal Validation in an OO Life-cycle of Protocols. *Int. workshop on Formal Methods in Object-Oriented Distributed Systems, FMOODS'96*, Paris, Mar. 1996. Chapman & Hall.
- [CI24] E. Fromentin, C. Jard, GV. Jourdan and M. Raynal.
Recognition of Patterns on Control Flows for Debugging Distributed Executions. *2nd International Workshop on Automated and Algorithmic Debugging*, May 22-24, St Malo, 1995.
- [CI23] C. Bareau, B. Caillaud, C. Jard and R. Thoraval.
Measuring Concurrency of Regular Distributed Computations, *TAPSOFT Conference*, Aarhus Denmark, May 1995, LNCS 915, pp. 711-725, Springer Verlag.
- [CI22] C. Jard.
Formal Methods to Observe Interoperability of Protocols, *Anglo-French workshop on formal methods, modelling and simulation for system engineering*, St Quentin France, Feb. 1995.

- [CI21] C. Jard.
How to Observe Interoperability at the Service Level of Protocols, *7th IFIP WG6.1 International Workshop on Protocol Test Systems (IWPTS'94)*, Nov. 1994, Tokyo, Japan.
- [CI20] T. Jéron and C. Jard.
3D Layout of Reachability Graphs of Communicating Processes, *Graph Drawing 94*, Princeton, Oct. 1994.
- [CI19] C. Jard and GV. Jourdan.
On the Coding of Dependences in Distributed Computations. *ACM PODC, Principles of Distributed Computing*, Los-Angeles, Aug. 1994.
- [CI18] C. Jard, T. Jéron, GV. Jourdan and JX. Rampon.
A General Approach to Trace-checking in Distributed Computing Systems. *IEEE ICDCS- 14th International Conference on Distributed Computing Systems*, Poznan, Jun. 1994.
- [CI17] C. Bareau, B. Caillaud, C. Jard and R. Thoraval.
Correctness of Automated Distribution of Sequential Programs. *PARLE Conference*, Munich, Jun. 1993, LNCS 694, pp. 517-528.
- [CI16] C. Diehl, C. Jard and JX. Rampon.
Reachability Analysis on Distributed Executions. *TAPSOFT Conference*, Paris Orsay, Apr. 1993, LNCS 668, pp. 629-643.
- [CI15] M. Andreu, M. Haziza, C. Jard and JM. Jézéquel.
Analysing a Space-protocol - from Validation to Experimentation (a Multi-method Approach). *FORTE Conference*, Perros Guirec, Oct. 1992.
- [CI14] C. Diehl and C. Jard.
Interval Approximations of Message Causality in Distributed Executions. *Symposium on the Theoretical Aspects of Computer Science. STACS*, Cachan, Feb. 1992, LNCS 577, pp. 363-376.
- [CI13] C. Jard and T. Jéron.
Bounded Memory Algorithms for Verification On-the-fly, *Proceedings of Computer-Aided-Verification, CAV91*, Aalborg, Danemark, Jul. 1991, LNCS 575, pp. 192-202.
- [CI12] T. Jéron and C. Jard.
Testing for Unboundedness of Fifo Channels, *Symposium on Theoretical Aspects of Computer Science, STACS*, Hambourg, Feb. 1991, LNCS 480, pp. 322-332.
- [CI11] C. Jard and T. Jéron.
On-line Model Checking for Linear Temporal Logic Specifications, *Proceedings of the Workshop on automatic verification methods for finite state systems*, Grenoble, Jun. 1989, LNCS 407 pp. 189-196.
- [CI10] C. Jard and JM. Jézéquel.
A Multiprocessor Estelle to C Compiler to Prototype Distributed Algorithms on Parallel Machines, *Proceedings of the 9th IFIP International Symposium on Protocol Specification, Testing, and Verification*, University of Twente, The Netherlands, North-Holland, Jun. 1989.
- [CI9] JM. Jézéquel, C. Jard.
Building a Global Time on Parallel Machines, *Proceedings of 3rd International Workshop on Distributed Algorithm*, Nice, France, Sep. 1989, Springer Verlag LNCS 392.
- [CI8] C. Jard and JM. Pageot.
Experience in Guiding Simulation, *Proceedings IFIP Symposium on protocols, PSTV*, Atlantic City, Jun. 1988.

[CI7] M. Adam, P. Ingels, C. Jard, JM. Jézéquel and M. Raynal.
Experimentation on Parallel Machines is Helpful to Analyse Distributed Algorithms, *Workshop internationale sur l'algorithme parallèle et distribué*, Bonas, Oct. 1988.

[CI6] C. Jard and M. Raynal.
The Rudiments of Object Distribution in Distributed Systems. *2th Int. Conf. on computers Inf. Sciences*, Istanbul, 1987.

[CI5] JM. Hélyary, C. Jard, N. Plouzeau and M. Raynal.
Detection of Stable Properties in Distributed Applications, *ACM Symposium on Principles of Distributed Computing (PODC)*, Vancouver, Canada, Aug. 1987.

[CI4] C. Jard, JF. Monin and R. Groz.
Experience in Implementing X250 in Veda, *IFIP Symposium on Protocols (PSTV)*, Toulouse, Jun. 1985.

[CI3] C. Jard, R. Groz and JF. Monin.
VEDA - a Software Simulator for the Validation of Protocol Specifications, *COMNET'85*, Hongrie, Oct. 1985.

[CI2] R. Groz, C. Jard and C. Lassudrie.
An Experience with complementary Validation Tools, *IFIP Symposium on Protocols (PSTV)*, New-York, Jun. 1984.

[CI1] Gv. Bochmann, E. Cerny, C. Jard and B. Sarikaya.
Some Experience with the Use of Formal Specifications, *IFIP Symposium on Protocols (PSTV)*, Idyllwild, May 1982.

V.5. Autres publications dans des colloques francophones

[CO15] L. Tanguy, C. Viho et C. Jard.
Un outil pour la parallélisation automatique de l'exécution répartie de cas de test, *CFIP'2000 : Colloque Francophone sur l'Ingénierie des Protocoles*, JP. Courtiat, M. Diaz, P. Sénac (ed), Hermès, p. 333-348, Toulouse, France, Oct. 2000.

[CO14] H. Canon et C. Jard.
Un modèle sémantique pour la validation des logiciels objets en télécommunication, *Colloque Francophone sur l'Ingénierie des Protocoles, CFIP 99*, Nancy, France, Hermès, A. Schaff (ed), p. 83-98, Avr. 1999.

[CO13] L. Héluët, C. Jard et B. Caillaud.
Une notion effective d'équivalence pour des familles de scénarios représentées par des HMSC, *Approches Formelles dans l'Assistance au Développement de Logiciel, Afadl'98*, Poitiers, 1998.

[CO12] A. Aghasaryan, A. Benveniste, R. Boubour, E. Fabre et C. Jard.
Suivi de pannes dans les réseaux de télécommunication - une approche par reconnaissance d'ordres partiels. *Actes du 2eme colloque francophone sur la gestion de réseau et de service (GRES'97)*, Rennes, Sep. 1997.

[CO11] C. Jard, T. Jéron, P. Morel et C. Viho.
Une expérience de génération automatique de tests pour un protocole de cohérence de caches. *Actes de la conférence TESTING'97*, Paris, Juin 1997.

[CO10] C. Jard et JM. Jézéquel.
Vers l'utilisation d'outils de validation dans un processus de conception objet de protocoles. *Actes des journées AFADL - Approches formelles dans l'assistance au développement de logiciels*, Toulouse, Mai 1997.

[CO9] R. Boubour et C. Jard.
Une approche pour des capteurs d'alarmes intelligents dans les réseaux. *Colloque Francophone sur l'Ingénierie des Protocoles*, Rabat, Oct. 1996.

[CO8] B. Caillaud, P. Caspi, A. Girault et C. Jard.

Un modèle pour la répartition d'automates réactifs sur réseau asynchrone de processeurs. *Colloque AFCET sur la modélisation des systèmes réactifs*, Brest, Mar. 1996.

[CO7] C. Jard.

Compilation d'algorithmes distribués sur machines parallèles, *Actes du colloque AFCET sur la programmation des machines parallèles*, Chamonix, Mai 1989.

[CO6] C. Jard et JM. Jézéquel.

Outils pour l'expérimentation d'algorithmes distribués sur machines parallèles, *Actes du Colloque C³ d'Angoulême*, Déc. 1988.

[CO5] C. Jard et M. Raynal.

Specification of Properties is Required to Verify Distributed Algorithms, *Colloque Afcet sur les langages de spécification*, fév. 1987.

[CO4] JM. Ayache, R. Groz, C. Jard et JF. Monin.

Des outils pour Estelle - du prototype au poste industriel, *Journées francophones sur l'informatique*, Liège, Janvier 1987.

[CO3] C. Jard.

Le projet Atlas d'algorithmes distribués, *Colloque C³*, Angoulême, Sep. 1985.

[CO2] R. Groz et C. Jard.

Estelle - un langage pour la description des protocoles, *Colloque Afcet sur les protocoles*, Toulouse, Avr. 1985.

[CO1] C. Jard, JF. Monin et R. Groz.

Evaluation et validation de protocoles - l'outil Veda, *Journées francophones de l'informatique*, Genève, Jan. 1985.

V.6. Conférences invitées dans le cadre de colloques

[CV25] L. Hérouet, C. Jard.

La manipulation formelle de scénarios pour les systèmes répartis : l'exemple des HMSC. *Modélisation des systèmes répartis*. G. Juanele, R. Valette (éditeurs), Hermes, Toulouse, Oct. 2001.

[CV24] C. Jard.

Using True-concurrency Models to Synthesize Tests from Formal Specifications. Proceedings of the Concordia Prestigious Workshop on Communication Software Engineering, R. Dssouli, F. Khendec (editors), Concordia university, Montréal, sep. 2001.

[CV23] C. Jard.

Executing Message Sequence Charts. *Dagstuhl seminar on Logic, Automata and Computer Science*, Dagstuhl, Nov. 2000.

[CV22] C. Jard.

Méthodes formelles et tests. *Trentenaire du génie logiciel*. Paris, Nov. 1999.

[CV21] C. Jard.

La recherche en tests de protocoles. *Journées pleinières SEE*, Paris-Toulouse, 1999.

[CV20] A. Benveniste, E. Fabre and C. Jard.

Partially Stochastic Petri Nets : a Framework for Fault Detection and Diagnosis in Large Distributed Systems, *Ecole MOVEP sur la modélisation des processus parallèles*, Nantes, Jul. 1998.

[CV19] C. Jard.

Automatic Test Generation Methods for Reactive Systems. *CIRM Summer School*, Marseille, 1998.

[CV18] C. Jard

Automatic Distribution of Test Cases. *Dagstuhl seminar on testing*, Dagstuhl, 1998.

[CV17] C. Jard.

Spécification, vérification et génération de tests pour des protocoles ATM. *Actes de la journée annuelle FORMA*, Paris, Janvier 1998.

[CV16] C. Jard.

Génération automatique de tests pour les protocoles. *Actes de la vingtième journée de l'OFTA*, Paris, Juin 1997.

[CV15] C. Jard.

Diagnostic dans les systèmes répartis fondé sur des modèles d'ordre partiel. *Actes de JBOPAD97*, Ordres partiels et algorithmes distribués, Bordeaux, Juin 1997.

[CV14] C. Jard.

Testing Behavioural properties of Systems using Model-based Verification Techniques. *FEmSys'97 - Formal Design of Safety Critical Embedded Systems*, Munich, April 1997.

[CV13] C. Jard.

On the Use of On-the-fly Verification to Generate Conformance Test Suites. *Application of Formal Methods to System Development - Telecommunications, VLSI and Real-Time Computerized Control Systems*, Montreal, Oct. 1996. Entretiens Jacques Cartier.

[CV12] C.Jard.

La génération de tests pour les protocoles à l'aide de la technologie de la vérification. *Colloque FORMA sur l'utilisation des méthodes formelles*, Grenoble, Janvier 1996.

[CV11] C. Jard.

Introduction to the Automatic Validation of Protocols. *EUNICE Summer School*, Lisbon, 26-30 June 1995. Invited Tutorial.

[CV10] C. Jard.

Observation des comportements en univers réparti. *Septièmes rencontres francophones du parallélisme*, Mons, Belgique, 30 mai-2 juin 1995.

[CV9] C. Jard and T. Jéron.

Formal Analysis of Distributed Computations, *Septièmes entretiens du centre Jacques Cartier, Communicating Informatics and Distributed Systems*, December 1994, Grenoble.

[CV8] C. Jard.

L'analyse des traces d'exécutions réparties. *Ecole d'informatique sur l'algorithmique répartie- calculs répartis et causalité*, Roscoff, Septembre 1993.

[CV7] C. Diehl et C. Jard.

Observation de la causalité dans les exécutions réparties. *Datation et contrôle des exécutions réparties, Bigre 78*, Mars 1992. Colloque C3.

[CV6] C. Jard.

The Event Lattice of Distributed Executions. *French-Israel workshop on distributed and parallel systems*, Grenoble, May 4-5, 1992.

[CV5] C. Jard and T. Jéron.

Depth-first Searches for on-line Verification by Buchi automata. *French-Israel workshop on formal verification for hardware and software*, Grenoble, Oct. 1991.

[CV4] C. Jard, B. Caillaud, JM. Jézéquel and N. Plouzeau.

Quelques expériences en matière de validation de protocoles industriels. Un point de vue de chercheur. *Actes du colloque CFIP sur l'ingénierie des protocoles*, Pau, Sep. 1991.

[CV3] F. André, C. Jard and JL. Pazat.

Data-driven Distribution of Sequential Programs on Distributed Machines, *Int. workshop on Compilers for Parallel Computers*, Paris, Dec. 1990.

[CV2] C. Jard.

Programming Distributed Computers- a Single Program Approach, *Book Eurocourses - Computing with Parallel Architectures*, Milan, Sep. 1990.

[CV1] C. Jard.

Valider les protocoles en simulant. *Actes du colloque CFIP'88*, Bordeaux, Sep. 1988, Eyrolles.

V.7. Documents techniques. Manuels

[RR2] JM. Jézéquel et C. Jard.

L'expérimentation d'algorithmes distribués sur machines parallèles avec ECHIDNA, *RR Irisa 603*, Sep. 1991, 60 pages.

[RR1] C. Diehl, C. Jard, H. Leroy, JL. Pazat, T. Priol et K. Wolinski.

Guide du débutant sur le TNode de l'Irisa. *RR Irisa 9101*, mars 1991. 60 pages.

V.8. Thèses

[TH1] C. Jard.

Protocoles et services - le test des spécifications. *Thèse de Docteur-Ingénieur en informatique de l'université de Rennes I*, Nu 173, mai 1984. Jury - J.P. Verjus (président), JP. Ansart, P. Azéma (rapporteurs), Gv. Bochmann, J.M. Pitié et M. Raynal. 140 pages.

[TH2] C. Jard.

Contribution à la vérification dynamique des protocoles. *Habilitation à diriger les recherches de l'université de Rennes I*, décembre 1994. Jury - J.P. Banâtre (président), Gv. Bochmann, JM. Pitié, J. Sifakis (rapporteurs), F. André, M. Raynal et J.P. Verjus. 150 pages, Coll. Irisa.