

Programme jeunes chercheuses et jeunes chercheurs 2005

APPEL A PROJETS DE RECHERCHE

A - FICHE D'IDENTITE DU PROJET

ATTENTION : Cette partie (A) sera à saisir directement sur le site de soumission

N°dossier : :JC05_43379:Baillot:Patrick:

(reprendre la référence qui vous sera attribuée automatiquement par le logiciel de soumission)

Secteur disciplinaire principal (*cf. liste en dernière page de ce dossier*) :

CSD 1- Sciences et Technologies de l'information et de la communication

Autre secteur disciplinaire facultatif (*cf. liste en dernière page de ce dossier*) :

Titre du projet (*maximum 120 caractères*)

Nouveaux Outils pour la Complexité : Sémantique et Types

Acronyme ou titre court (*12 caractères*) **NO-COST**

Mots-clés (la liste des mots-clés sera donnée sur le logiciel de soumission)

Qualité-Sécurité du logiciel, Algorithmie

Responsable du projet :

- M ou Mme : M Prénom : Patrick

Nom : BAILLOT

- Date de naissance : 23/07/1972

- Emploi actuel : CR2 au CNRS.

- Date de nomination dans cet emploi : 1/10/2001

- Section CNU ou CNRS ou Inserm ou équivalent : section 7 du CNRS

- Téléphone : (+33) 1 49 40 40 67 - Télécopie : (+33) 1 48 26 07 12

- Adresse électronique : patrick.baillot@lipn.univ-paris13.fr

Laboratoire :

- N° Unité : UMR 7030

- Laboratoire ou équipe (nom complet) : Laboratoire d'Informatique de l'université Paris-Nord

- Directeur : Christophe Fouqueré

- Adresse complète :

Laboratoire d'Informatique de l'université Paris-Nord

UMR CNRS 7030

Institut Galilée – Université Paris-Nord

99, avenue Jean-Baptiste Clément

93430 Villetaneuse

Durée du projet : 36 mois

Montant total de l'aide demandée : 195,7 KEuros

(reporter ici la valeur W de la fiche C)

Estimation (pour information) du coût complet de la demande : 584,2 KEuros

(reporter ici la valeur X de la fiche C)

Résumé du projet (*maximum 3000 caractères*)
(objectifs, résultats attendus, méthodologie)

La complexité implicite étudie des calculs et des langages dont les programmes ont une complexité bornée, en terme de temps de calcul ou de mémoire requise (polynomial, exponentiel, ... par rapport aux données). Plusieurs approches ont été utilisées pour garantir de telles bornes, notamment la Logique linéaire et ses variantes, les systèmes de réécriture avec quasi-interprétations, les types linéaires avec ressources pour les langages fonctionnels. Cette théorie nécessite désormais que l'on trouve des éléments unificateurs, soit par des systèmes plus généraux soit par des approches plus sémantiques. La sémantique des jeux modélise les langages de programmation en interprétant les types par des jeux à deux joueurs, les programmes par des stratégies pour l'un des joueurs (l'autre joueur représentant l'environnement d'évaluation) et les interactions par des parties. Cette présentation explicite de l'interaction comme suite de coups devrait permettre de parler de complexité au niveau sémantique.

Objectifs :

- A. une approche sémantique unificatrice pour les divers systèmes de complexité implicite à travers la sémantique des jeux;
- B. l'extension des résultats concernant les systèmes connus, notamment la logique linéaire;
- C. de nouveaux systèmes plus généraux, voire un cadre unitaire recouvrant plusieurs des systèmes actuels de complexité implicite.

Résultats attendus :

D'une part des résultats fondationnels, en particulier une sémantique des jeux polynomiale permettant de comparer les systèmes de complexité implicite existants.

D'autre part l'amélioration de l'utilisation des systèmes de complexité implicite pour la vérification statique de complexité, notamment grâce aux systèmes de types.

Enfin la définition de nouveaux systèmes pourrait permettre d'obtenir des méthodes de vérification plus flexibles.

Méthodologie :

A. Sémantique.

On définira une sémantique des jeux équipée d'une notion de complexité pour les stratégies, puis une sémantique de stratégies polynomiales. Dans un second temps on interprétera les systèmes de complexité implicite polynomiale dans cette sémantique de jeux.

B. Types lambda-calcul et réseaux.

On poursuivra l'étude des variantes de logique linéaire à complexité bornée comme systèmes de types pour le lambda-calcul : on s'intéressera à l'inférence de type, aux applications à la réduction optimale et aux extensions avec des types récursifs.

C. Complexité implicite et expressivité.

Utilisant les résultats des parties A et B on cherchera à définir un système plus général englobant plusieurs des systèmes existants, par exemple les types linéaires d'Hofmann et un système de logique linéaire polynomiale. Enfin on considérera la complexité au sens du modèle BSS sur une structure arbitraire et l'adaptation à ce cadre des systèmes de complexité implicite polynomiale avec ordre supérieur.

Abstract (Not exceed 3000 car.)
(objectives, expected results, methodology)

Implicit computational complexity studies calculi and languages for which programs have a bounded complexity, in terms of computation time or of memory usage (polynomial, exponential ... in the size of the input). Various approaches have been used to ensure such bounds, in particular: Linear logic and its variants, rewriting systems and quasi-interpretations, linear types with resource types for functional languages. This theory would now need unifying elements, either with more general systems or with more semantical approaches.

Game semantics modelizes programming languages by interpreting types by two player games, programs by strategies for one of the players (the other one representing the evaluation environment) and interactions by plays. This explicit representation of interaction as a sequence of moves should allow to handle complexity at the semantic level.

Objectives:

- A. a semantic unifying approach for various systems of implicit computational complexity by game semantics;
- B. extension of the results dealing with the existing systems, in particular linear logic;
- C. new and more general systems, or even a unifying framework subsuming several of the existing implicit complexity systems.

Expected results:

- Foundational results, in particular a semantics of polynomial games allowing to compare the existing systems of implicit complexity;
 - Improvement of the use of implicit complexity systems for static verification of complexity, in particular with type systems.
- Finally the definition of new systems could allow to obtain more flexible verification methods.

Methodology:

A. Semantics.

We will define a game semantics equipped with a notion of complexity for strategies, then a semantics of polynomial strategies. In a second step we will interpret polynomial implicit computational systems in this semantics.

B. Types, lambda-calculus and proof-nets.

We will proceed with the study of variants of Linear logic with bounded complexity as type systems for lambda-calculus: we will be interested in type inference, in the applications to optimal reduction and in the extension with recursive types.

C. Implicit complexity and expressivity.

Using the results of parts A and B we will try to define a more general system subsuming several of the existing systems, for instance Hofmann's linear types and a system of polynomial linear logic. Finally we will consider complexity in the BSS model over an arbitrary structure and the adaptation to this setting of the systems of implicit complexity with higher-order.

Programme jeunes chercheuses et jeunes chercheurs 2005

Participants au projet

L'équipe proposée devra être composée majoritairement de jeunes chercheuses et de jeunes chercheurs titulaires d'un emploi permanent, nommément identifiés pour lesquels l'implication dans le projet devra représenter une partie importante de leur activité de recherche ; cette implication devra être majeure, voire exclusive, en ce qui concerne le ou la responsable du projet. Pour les enseignants-chercheurs, il s'agit du pourcentage de leur temps de recherche. *Indiquez en premier le nom du responsable du projet.*

Nom	Prénom	Laboratoire ou équipe de rattachement	Emploi actuel	Date de nomination dans le poste	Age au 1/1/05	% du temps de recherche consacré au projet
Baillot	Patrick	LIPN	CR2, CNRS	1/10/2001	32	80%
Boudes	Pierre	LIPN ¹	MCF	1/09/2005	30	50%
Laurent	Olivier	PPS	CR2, CNRS	1/10/2001	27	40%
Mogbil	Virgile	LIPN	MCF	1/09/2002	33	50%

L'équipe est renforcée dans la thématique de ce projet par :
un post-doctorat CNRS Maths-STIC de 12 mois au LIPN pour 2005-06 (Jean-Yves Moyen à *confirmer*),
un post-doctorat Université Paris 13 de 3 mois au LIPN en 2005-06 (Ugo Dal lago à *confirmer*).

Ainsi que les demandes suivantes toujours sur cette thématique :
deux allocations au titre de l'école doctorale de 3 ans au LIPN pour 2005-2008 (Vincent Atassi , Jean-Baptiste Blanquis)
un demi-ATER à l'université Paris 13 pour 2005-06 (Paulin Jacobé de Naurois).

Auxquelles nous souhaiterions ajouter une demande d'allocation dans le cadre de ce projet.²

Pour chacun des membres de l'équipe fournir une biographie **d'une page maximum** qui comportera :

A/ Nom, prénom, âge, doctorat, stage post-doctoral, situation actuelle

B/ Autres expériences professionnelles

C/ Liste des 5 publications (ou brevets) les plus significatives des cinq dernières années

D/ Prix, distinctions

Signature du responsable du projet :

¹ Pierre Boudes prend ses fonctions à la rentrée 2005 au LIPN. Il est stagiaire en 2005-2006.

² L'équipe dans laquelle sont les membres du LIPN compte des professeurs dans une thématique plus générale et susceptibles d'assumer un encadrement de doctorants. De plus Patrick Baillot souhaite passer une habilitation à diriger les recherches entre 2005 et 2006.

Patrick BAILLOT
patrick.baillot@lipn.univ-paris13.fr
<http://www-lipn.univ-paris13.fr/~baillot/>
né le 23/07/1972. (32 ans)

Situation actuelle : chargé de recherche au CNRS (CR2), affecté au LIPN (UMR7030), depuis 1/10/2001.

Situations antérieures : 9/2000-9/2001 ATER à l'ES2, Marseille, membre du LIF.

Stage Post-doctoral : 9/1999-9/2000 université d'Edimbourg (UK), Laboratory for Foundations of Computer Science (LFCS), sous la direction de S.Abramsky, financé par réseau européen TMR Linear.

Doctorat: soutenu le 22/1/1999, université Aix-Marseille II math. discrètes et fondements de l'informatique. Titre : *Approches dynamiques en sémantique de la logique linéaire : jeux et géométrie de l'interaction*. Directeur: J.-Y. Girard

Autres expériences professionnelles:

enseignement :

2003-04 cours de Théorie de la démonstration au DEA de Logique de l'université Paris 7, participation de 2001-03.

encadrement de stages de DEA/Master2 :

2002 : D.de Carvalho (DEA de Logique Paris 7) ; 2005 : V.Atassi (Paris 6 et MPRI)

co-encadrement de thèse : D. de Carvalho, université Aix-Marseille II. Directeur :

T.Ehrhard, allocation de thèse associée à l'ACI GEOCAL. Depuis 9/2003.

Participation à des projets de recherche:

comme responsable de site:

2003-06 : Projet ACI NIM, GEOCAL (Géométrie du Calcul) resp.: T. Ehrhard.

2003-06 : Projet ACI SI, CRISS (Contrôle de Ressources et d'Interférences dans les Systèmes Synchrones) resp.: R. Amadio.

2002-03 : Action Spécifique CNRS Méthodes Formelles pour la Mobilité, resp.: R. Amadio.

comme co-responsable de programmes d'échanges bi-nationaux:

2002 : programme Vinci France-Italie (université Franco-Italienne)

2004-05 : coopération CNRS-CNR France-Italie (2 ans). Resp. italien: M.Pedicini.

2005 : responsable d'un projet de Post-Doc CNRS STIC (STIC 6, 12 mois) accepté.

(titre: Caractérisation de la complexité polynomiale en temps par les sémantiques de jeux).

Publications (5 publications les plus significatives des 5 dernières années) :

P.Baillot. Type inference for light affine logic via constraints on words
Computer Science , 328(3):289-323, décembre 2004.

Theoretical

P.Baillot. Stratified coherence spaces: a denotational semantics for Light Linear Logic
Theoretical Computer Science , 318 (1-2), 2004.

P.Baillot, M.Pedicini. Elementary complexity and geometry of interaction
Fundamenta Informaticae vol. 45,no 1-2, 2001.

P.Baillot, K.Terui. Light types for polynomial time computation in lambda-calculus
Proc. of International Symp. on Logic in Computer Science (LICS'04) pp. 266-275, 2004.

P.Baillot, V.Mogbil. Soft lambda-calculus: a language for polynomial time computation
of FoSSaCS'04, volume 2987 of LNCS, pp.27-41, Springer.

Proc.

Pierre BOUDES

pierre.boudes@lipn.univ-paris13.fr

<http://www-lipn.univ-paris13.fr/~boudes/>

né le 20/04/1974 (31 ans)

Situation actuelle : Ingénieur en informatique, *Arboost Technologies*, Marseille. Recherche et développement en sûreté de fonctionnement.

À la rentrée 2005 : Maître de conférence en informatique à l'université Paris 13, membre du LIPN (UMR 7030).

Situations antérieures :

2002-03 : ATER en informatique, université de Provence, Marseille.

2001-02 : ATER en mathématiques, université de la Méditerranée, Marseille.

Doctorat : soutenu le 20/12/2002, université Aix-Marseille II, mathématiques discrètes et fondements de l'informatique. Titre : *hypercohérences et jeux*. Directeur : Thomas Ehrhard.

Autres expériences professionnelles :

enseignement :

1999 encadrement d'un mémoire de maîtrise (TER) de mathématiques sur les fonctions calculables en logique linéaire élémentaire.

participation à des projets de recherche :

2003-04 : ACI NIM, Géométrie du Calcul (GéoCal), non-permanent.

2004-05 : Projet CNRS-CNR (coopération France-Italie 2ans) Interaction et Complexité.

Publications :

P. Boudes, Non-uniform hypercoherences

Electronic notes in theoretical computer science, volume 69, Elsevier 2003

categorical theory and computer science CTCS 2002.

P. Boudes, Projecting games on hypercoherences (polarized bordered games)

Lecture Notes in Computer Science volume 3142, Springer, international colloquium on automata, languages and programming ICALP 2004.

Olivier LAURENT
Olivier.Laurent@pps.jussieu.fr
<http://www.pps.jussieu.fr/~laurent/>
né le 11/06/1977 (28 ans)

Situation actuelle : chargé de recherche au CNRS (CR2), affecté au laboratoire PPS (UMR 7126), depuis le 1/10/2001.

Doctorat : soutenu le 11/03/2002, univ. Aix-Marseille II, mathématiques discrètes et fondements de l'informatique. Titre : *Étude de la polarisation en logique*. Directeurs : Jean-Yves Girard et Laurent Regnier.

Autres expériences professionnelles :

enseignement (depuis 2001) :

cours de Sémantique des jeux en DEA d'informatique.
cours de Théorie de la démonstration en DEA-Master de mathématiques.
encadrement de 3 mémoires de DEA et co-direction d'une thèse en cours.

responsabilités (depuis 2001) :

membre du conseil de laboratoire et de la commission des moyens de PPS.
responsable du site internet et des prépublications de PPS.
membre du conseil du master 2 de logique math. et fondements de l'informatique.
membre des commissions de spécialistes section 25 à Paris 7, section 25-26 à Aix-Marseille 2, section 27 à Paris 13.

participation à des projets de recherche (commencés après 2001) :

2002-03 : Projet CNRS Math-STIC Logique et sémantique des langages de programmation.
2002-03 : Action Spécifique CNRS Méthodes Formelles pour la Mobilité.
2003-07 : Réseau Thématique du programme européen Information Society Technologies, projet Applied Semantics (AppSem)
2003-06 : Projet ACI NIM, Géométrie du Calcul (GéoCal).
2004-05 : Projet CNRS-CNR (coopération France-Italie 2ans) Interaction et Complexité.

Publications (sélection de 5) :

O. Laurent, Polarized proof-nets and lambda-mu calculus.
Theoretical Computer Science, 290(1):161-188, 2003.

O. Laurent and L. Regnier, About Translations of Classical Logic into Polarized Linear Logic.
Proc. Logic In Computer Science (LICS), 2003.

O. Laurent and L. Tortora de Falco, Slicing polarized additive normalization.
Linear Logic in Computer Science (London Mathematical Society Lecture Notes Series 316 - CUP), 2004.

O. Laurent, Polarized games.
Annals of Pure and Applied Logic, 130(1-3):79-123, 2004.

O. Laurent, Classical isomorphisms of types.
To appear in *Mathematical Structures in Computer Science*, 2005.

Virgile MOGBIL
virgile.mogbil@lipn.univ-paris13.fr
<http://www-lipn.univ-paris13.fr/~mogbil>
né le 16/07/1971 (33 ans).

Situation actuelle : maître de conférence section 27 au LIPN (UMR7030) depuis 1/09/2002.

Situations antérieures : 9/1999-9/2001: ATER informatique à l'université de Provence, Marseille.

Stage post-doctoral : 10/2001-3/2002 Università degli studi Roma Tre et Università La Sapienza, Rome, Italie, sous la direction de M. Abrusci, financé par réseau européen TMR Linear.

Doctorat : soutenu le 17/1/2001, université Aix-Marseille II math. discrètes et fondements de l'informatique. Titre : *Sémantique des phases, réseaux de preuve et divers problèmes de décision en logique linéaire*. Directeur : Yves Lafont.

Autres expériences professionnelles :

Enseignements à l'université Paris 13 (depuis nomination) :

(notations LMD, et ISPG pour école d'ingénieurs en informatique)

2004-05 : bases de données ISPG1, projet M1

2002-05 : sémantique ISPG1, algorithmique L3, ISPG1.

2003-04 : logique linéaire et applications DEA, projet M1

2002-04 : informatique L1, L2, bases de données M1, stages ISPG3

2002-03 : stage DEA : R. De Vismes (DEA de Logique Paris 7).

Responsabilités à l'université Paris 13 (depuis nomination) :

Responsable d'un programme Socrates avec l'Università degli Studi Roma Tre, Italie, dans le cadre d'un accord bilatéral ERASMUS, depuis 2003.

Responsable du Master mention Informatique 1ère année, depuis 2004.

Membre du bureau du département informatique, depuis 2004.

Membre élu de la commission de spécialistes section 27, depuis 2004

Participation à des projets de recherche (commencés après 2002) :

2003-06 : Projet ACI NIM : GEOCAL (Géométrie du Calcul) resp.: T. Ehrhard.

2003-06 : Projet ACI SI : CRISS (Contrôle de Ressources et d'Interférences dans les Syst. Synchrones) resp.: R. Amadio.

2004-05 : (2 ans) Projet CNRS-CNR (coopération France-Italie) Interaction et Complexité.

Publications avec comité de lecture dans des revues/conférences internationales :

C.Fouqueré, V.Mogbil, Rewritings in polarized (partial) proof structures, *preprint LIPN* numéro 2004-05, Structures and Deduction, Lisbon, Portugal, July 2005.

P.Baillet, V.Mogbil, Soft lambda-calculus: a language for polynomial time computation, *Lecture Notes in Computer Science* 2987, pp. 27-41, proc. of the 7th International Conference Foundations of Software Science and Computation Structures, 2004.

T.Krantz, V.Mogbil, Encoding hamiltonians circuits into multiplicative linear logic, *Theoretical Computer Science*, 266 (1-2) pp. 987-996, Elsevier, 2001.

V.Mogbil, Quadratic correctness criterion for Non commutative Logic, *Lecture Notes in Computer Science* 2142, pp. 69-83, proc. of the 10th Annual Conference of the EACSL, 2001.

Programme jeunes chercheuses et jeunes chercheurs 2005

B - Description du projet

N°dossier : :JC05_43379:Baillot:Patrick:

(reprendre la référence qui vous sera attribuée automatiquement par le logiciel de soumission)

Acronyme ou titre court du projet : **NO-COST**

B-1 – Equipe : (1 page maximum en arial 11, simple interligne)

On décrira ici l'histoire de la constitution de l'équipe, la place de l'équipe dans le laboratoire ou l'unité de recherche, le degré d'autonomie escompté, la complémentarité entre les membres et leur compétence dans le domaine.

L'idée de ce projet a une double origine :

D'un point de vue scientifique : un besoin se fait sentir en complexité implicite de renouveler cette branche de recherche avec des bases sémantiques plus solides. La sémantique des jeux apparaît particulièrement bien appropriée.

D'un point de vue d'équipe : l'équipe *Logique, Calcul et Raisonnement* du LIPN dispose d'un axe solide en Logique linéaire ; elle a en outre bénéficié de plusieurs recrutements ces dernières années, qui au sein du groupe de logique ont renforcé notamment les directions de la programmation fonctionnelle d'une part et de la complexité d'autre part. L'arrivée de Pierre Boudes comme MCF en septembre 2005 apporte des compétences nouvelles en sémantique. Ceci est l'occasion de joindre des forces complémentaires sur un projet nouveau qui, bien que lié à la logique linéaire, est organisé autour d'un objectif spécifique : comprendre et contrôler la complexité des programmes au moyen de la sémantique et de la logique.

Par ailleurs un post-doctorat CNRS (de 12 mois) a été attribué par le département STIC du CNRS sur le thème *Caractérisation de la complexité polynomiale en temps par les sémantiques de jeux* (septembre 2005). L'année 2005-06 nous semble donc un moment particulièrement propice pour lancer un projet sur ce thème, qui devrait ainsi bénéficier d'une forte synergie.

Il faut en outre noter que deux stagiaires travaillent actuellement en logique linéaire et complexité au LIPN (V. Atassi en Master 2 et J.-B. Bianquis en 4^e année d'ENS) et souhaiteraient entamer une thèse dans ce domaine. Enfin, une session résidentielle au CIRM³ devrait être organisée en février 2006 (sous réserve d'acceptation) par l'ACI GEOCAL et inclura notamment une école thématique (3 jours) sur *Complexité et Logique* ainsi qu'un workshop sur ce même thème (2 jours) et un workshop sur la sémantique. Olivier Laurent et Patrick Baillot participent à l'organisation de cette session, qui devrait être pour notre projet, alors dans sa première année, une rencontre très utile en permettant un état de l'art et un tour d'horizon des questions ouvertes dans les thèmes qui nous concernent.

Sur la composition de l'équipe soulignons que :

Olivier Laurent et Pierre Boudes apportent des compétences en sémantique des jeux et en sémantique dénotationnelle ;

Patrick Baillot et Virgile Mogbil ont de solides expériences concernant les liens entre logique linéaire et complexité.

Le laboratoire PPS dont est membre Olivier Laurent a des liens suivis avec le LIPN (notamment au travers de la participation à des projets).

Une collaboration entre Virgile Mogbil et Patrick Baillot est menée sur la logique linéaire et la complexité polynomiale depuis 2003, notamment dans le cadre de l'ACI CRISS. Des collaborations avec des chercheurs étrangers ont aussi été menées sur ce sujet, avec K.Terui (Tokyo) et U.Dal Lago (Bologne).

Enfin, ces 4 participants se connaissent bien, en particulier car ils ont effectué leur thèse à Marseille. P.Boudes et Olivier Laurent entretiennent des contacts scientifiques réguliers sur le thème des jeux. O.Laurent et Patrick Baillot ont assuré ensemble un cours de *Théorie de la démonstration* au sein du DEA de Logique de l'université Paris 7 en 2003-04.

³ Centre International de Rencontres Mathématiques (Marseille).

B-2 – Description du projet et résultats attendus : (8 pages maximum en arial 11, simple interligne)

On décrira les objectifs scientifiques du projet dans le contexte international, l'originalité par rapport aux thèmes du laboratoire, la méthodologie envisagée, l'interdisciplinarité éventuelle du projet, le déroulement prévisionnel et les diverses phases intermédiaires ainsi que les méthodologies employées. Des collaborations avec des partenaires extérieurs peuvent être ajoutées si elles apportent une compétence nécessaire à la mise en œuvre du projet. Les moyens demandés devront être en accord avec les objectifs scientifiques du projet.

1. Introduction

Les méthodes formelles (typage, preuve assistée...) permettent à des niveaux divers de garantir pour les logiciels un certain nombre de propriétés : absence d'erreurs à l'exécution, correction vis-à-vis d'une spécification, terminaison ... Cependant un aspect des programmes essentiel en pratique est l'usage qu'ils font des ressources, temps et espace. Ces aspects quantitatifs sont représentés par la complexité calculatoire du programme. On s'intéresse souvent alors à des classes de programmes, comme ceux terminant en temps polynomial. Alors qu'une branche de l'algorithmique se consacre à l'analyse de complexité des programmes on souhaiterait, du côté de la sémantique et de l'étude des langages de programmation, disposer d'outils pour analyser les classes de programmes présentant certaines bornes de complexité.

Il ne s'agit pas ici de méthode d'analyse fine au cas par cas de la complexité exacte d'un programme, mais de déterminer certaines sous-classes de langages ou de logiques qui, en se limitant à certaines disciplines de programmation, offrent la garantie d'une borne de complexité a priori. Ce domaine, celui de la complexité implicite, met à profit notamment des outils de logique ([Girard98]), théorie de la récursivité ([BellantoniCook92,LeivantMarion93,Leivant94,BNS00]) ou des systèmes de réécriture ([MarionMoyen00,BMM04]).

Une des approches en complexité implicite est basée sur la logique linéaire: celle-ci donne une place explicite à la notion de ressource et s'inscrit dans la correspondance preuves-programmes, sous-jacent au paradigme de la programmation fonctionnelle (langages CAML, Haskell). La logique linéaire permet par exemple de définir des systèmes de types pour les programmes fonctionnels garantissant qu'un programme termine en temps polynomial.

2. Objectifs

La complexité implicite est un thème encore récent présentant des techniques et des approches diverses et pas de théorie unificatrice. Nous pensons que les progrès vers une utilisation plus effective de ces méthodes nécessitent auparavant un travail amont et exploratoire. Le but de ce projet est de mettre en place un tel travail et de le développer, tout en améliorant les méthodes existantes et en les expérimentant. Pour cela nous distinguons les objectifs suivants :

- (A) Donner un cadre sémantique solide à l'étude de la complexité des programmes fonctionnels, plus particulièrement pour la classe des programmes de complexité polynomiale.
- (B) Poursuivre l'étude de la complexité implicite par les systèmes de logiques linéaires existants et étendre les résultats dans ce domaine.
- (C) Définir de nouvelles caractérisations logiques de la complexité polynomiale et d'autres classes, offrant plus d'expressivité en termes de programmes et/ou de meilleures procédures d'analyse.

L'objectif (A) est une direction originale et pour laquelle nous souhaitons utiliser le cadre des sémantiques de jeux. L'objectif (B) se situe dans des lignes de recherche déjà en activité mais pour lesquelles beaucoup de questions importantes sont encore ouvertes. Enfin, concernant le point (C), nous pourrions utiliser deux approches complémentaires: l'approche sémantique issue de (A) pourrait donner les moyens de définir un cadre unitaire recouvrant plusieurs systèmes actuels de complexité polynomiale; de façon complémentaire en confrontant les approches de logique linéaire à d'autres approches plus flexibles en complexité implicite, basées sur la réécriture ou le typage, on pourra chercher à définir de nouveaux systèmes offrant de meilleures propriétés.

3. Description

3.1 Contexte :

Le domaine de la complexité implicite vise à définir des calculs ou des langages de programmation où tous les programmes ont par construction une borne de complexité donnée, par exemple polynomiale en temps (Ptime). Les approches utilisées empruntent entre autres à la Logique linéaire ([Girard98]), la théorie de la récursion ([BellantoniCook92]), au typage ([Hofmann99]) ou aux systèmes de réécriture ([MarionMoyen00]). La classe la plus étudiée est la classe Ptime.

* *Logique linéaire.*

La logique linéaire (LL) ([Girard87]) est une logique accordant une place explicite à la notion de ressource. Une manière de la voir est comme un système de types pour le lambda-calcul, le noyau des langages fonctionnels. Ce système de types permet de distinguer entre une fonction utilisant son argument exactement une fois et une fonction l'utilisant un nombre arbitraire de fois, ceci grâce à une modalité spécifique pour la réutilisation d'argument (notée « ! »). Avec la modalité ! un statut logique est donc donné à la duplication.

Une autre manière de la voir est à travers la correspondance de Curry-Howard entre preuves et programmes: une procédure de réécriture des preuves (élimination des coupures) correspond à l'exécution des programmes. La logique linéaire raffine la logique intuitionniste (logique sans tiers exclu): il existe une traduction de cette dernière dans LL. Enfin, les preuves de la LL disposent d'une syntaxe sous forme de graphe, les réseaux de preuves, qui permettent une étude géométrique de l'élimination des coupures.

Des variantes à complexité bornée de LL sont obtenues en considérant des règles plus restrictives pour la modalité ! : c'est le cas des Logiques linéaires Bornées ([GSS92]), Light (LLL, [Girard98,AspertiRoversi02]) et Soft (SLL, [Lafont04]), dans lesquelles l'élimination des coupures est polynomiale. Une autre logique, la Logique linéaire élémentaire, correspond de la même manière à la complexité élémentaire (tour d'exponentielles de hauteur bornée).

A travers la correspondance de Curry-Howard ces logiques peuvent être vues comme des systèmes de types pour le lambda-calcul: si un terme est bien typé, alors il correspond à une fonction Ptime (resp. élémentaire). L'inférence de type est décidable ([Baillot04, CoppolaMartini01]) et peut ainsi être utilisée pour vérifier qu'un programme est Ptime (resp. élémentairement récursif).

Les systèmes LLL et SLL sont complets pour le temps polynomial dans le sens où ils permettent de représenter exactement les fonctions Ptime. Cependant pour comparer les différents systèmes de complexité implicite il n'est pas suffisant de considérer uniquement les *fonctions* qu'ils permettent de représenter. En effet celles-ci nous renseignent sur l'*expressivité extensionnelle* du système, mais pas sur son *expressivité intensionnelle*, c'est-à-dire sur les algorithmes qu'il permet de programmer. Ainsi par exemple un système de complexité Ptime permettra de représenter la fonction de tri mais pas nécessairement au moyen de l'algorithme quicksort.

Un autre critère de comparaison est ce qu'on pourrait appeler la *robustesse* de l'approche : avec quels traits de programmation ou de raisonnement peut-elle être étendue tout en conservant ses propriétés ?

Si on s'intéresse à la complexité d'un point de vue théorique, afin par exemple de comparer des classes de complexité ou de donner des bornes de complexité pour des problèmes, alors l'accent est mis sur l'expressivité extensionnelle. Inversement dans une perspective de vérification ou de certification de complexité de programmes on s'intéresse plutôt à l'expressivité intensionnelle, qui justifie l'intérêt du système considéré en pratique, et à sa robustesse, qui laisse la possibilité de généralisations.

Les systèmes de Logique linéaire Light et Soft offrent une expressivité intensionnelle assez limitée mais sont assez robustes dans le sens où elles intègrent le polymorphisme et les fonctions d'ordre supérieur.

Inversement d'autres systèmes présentent une expressivité extensionnelle différente ou moins de robustesse, mais des avantages par rapport à l'intensionnalité.

* *Systèmes de types linéaires, quasi-interprétations.*

A l'inverse de l'approche de la logique linéaire, qui repose sur un contrôle des fonctions (contrôle du nombre d'utilisations d'argument), d'autres approches s'appuient plus directement sur un contrôle

de l' *espace de calcul* , avec des techniques différentes: c'est le cas d'une part du système de types linéaires LFPL de Hofmann ([Hofmann99]), d'autre part des systèmes de réécriture avec *quasi-interprétations* polynomiales de Marion et Moyen ([MarionMoyen00]).

Le système LFPL est un système de types pour un mini-langage fonctionnel avec récursion structurelle. Il s'agit d'un système de types linéaires, c'est-à-dire basé sur la logique linéaire, mais sans modalité, et avec un type de base spécifique représentant une ressource mémoire (une unité mémoire). Le type d'une fonction précise alors l'allocation mémoire qui lui est nécessaire et on programme sans allocation dynamique. Ce système permet de rendre compte des calculs en espace linéaire et (à la fois) en temps polynomial. Même s'il ne capture pas toutes les fonctions Ptime, LFPL permet de représenter une bonne variété d'algorithmes courants, c'est-à-dire que son expressivité intensionnelle est assez large. En revanche il ne dispose pas de polymorphisme et ne permet pas la duplication de fonction, des caractéristiques qui sont présentes dans les approches par la logique linéaire.

L'approche des *quasi-interprétations* développée par Marion et Moyen se place, elle, dans le cadre des systèmes de réécriture du 1^{er} ordre. Une quasi-interprétation polynomiale pour un tel système assigne une fonction majorée par un polynôme à chaque symbole et doit être décroissante (au sens large) vis-à-vis des règles de réduction. Couplées avec des techniques d'ordres de terminaison, les quasi-interprétations donnent une caractérisation des fonctions Ptime (et d'autres classes de complexité). Cette approche offre aussi une large expressivité intensionnelle et a l'avantage d'être compatible avec des définitions récursives (et pas uniquement de la récursion structurelle).

* *Sémantique des jeux*:

Traditionnellement la sémantique dénotationnelle interprète les programmes comme des fonctions ou des relations et n'en retient ainsi que l'aspect entrée-sortie: les propriétés comportementales des programmes sont alors reflétées par des propriétés mathématiques des fonctions (continuité, stabilité). On a obtenu plus de latitude avec l'introduction dans les années '90 des sémantiques de jeux ([AJM00,HylandOng00]), dans lesquelles l'interaction entre programme et environnement est représentée par une *partie*, c'est-à-dire une suite alternée de coups entre 2 protagonistes (appelés Opposant et Joueur). Les fonctions y sont alors remplacées par des *stratégies* pour Joueur. On peut comprendre les parties comme des traces d'interaction où le temps est explicite. Ce cadre des jeux a permis de modéliser et de rendre compte d'une large variété de langages de programmation avec noyau fonctionnel, démontrant ainsi sa pertinence et sa souplesse: langages avec contrôle (exceptions) ([Laird97,Laurent04]), aspects impératifs ([AbramskyMcCusker99]), non-déterminisme ([Harmer99])... Il commence à être utilisé pour la vérification de logiciel ([AGMO04]), pour laquelle sa propriété de compositionnalité est un atout.

3.2 Projet :

A. Sémantique (Patrick Baillot, Pierre Boudes, Olivier Laurent, [Daniel de Carvalho, Ugo Dal Lago]⁴)

Nous souhaitons utiliser la sémantique dénotationnelle comme un cadre général pour étudier la complexité des programmes fonctionnels. Nous tirerons plus spécifiquement parti des sémantiques de jeux car elles semblent propices à définir le temps d'une interaction. Partant de là nous pouvons envisager de considérer la complexité d'une stratégie et par conséquent des classes de stratégies correspondant à des classes de complexité.

Des pistes pour parvenir à définir la complexité des stratégies pourraient venir de la géométrie de l'interaction (GdI) ([Girard88,Laurent01]) : cette branche de recherche décrit l'exécution des programmes par le calcul de chemins dans leur représentation comme arbre syntaxique ou réseau. La GdI est issue de la Logique linéaire et a été reliée à la complexité notamment dans [BaillotPedicini01,DalLago05]. Enfin la GdI est très proche de certains modèles de jeux (jeu d'Abramsky-Jagadeesan-Malacaria [AJM00]). Par ailleurs, des sémantiques intermédiaires entre les modèles de jeux et les modèles de fonctions ont été étudiées ([BDER97,Boudes04,Boudes05]) et pourraient permettre de rendre compte d'aspects quantitatifs du calcul sans pour autant traiter le temps de façon aussi explicite que dans les jeux. Cette direction fournit une deuxième approche intermédiaire possible pour appréhender la complexité dans les sémantiques de jeux.

4 Non permanents entre [...].

Il serait particulièrement intéressant de disposer d'un modèle de jeux dans lequel toutes les stratégies représentent des algorithmes de temps polynomial. Un tel modèle pourrait être légitimement considéré comme une sémantique intrinsèquement polynomiale. Il constituerait un point de départ solide pour plusieurs directions de recherche :

Comparaisons : il existe actuellement des caractérisations du temps polynomial basées sur des principes très différents: logique linéaire, restrictions de la récursion primitive, contrôle de l'espace de calcul. Il en sera question plus en détail dans les parties B et C. Les comparaisons des ces différents systèmes, en particulier les traductions de l'un vers l'autre, se sont révélées jusqu'à présent des problèmes étonnamment difficiles. Un modèle de jeux polynomial pourrait fournir un bon cadre commun pour leur étude comparée. Pour cela on pourrait interpréter ces systèmes dans le modèle, ce qui fournirait une nouvelle démonstration de leur borne polynomiale et offrirait un même environnement pour les comparer. On peut noter qu'un modèle de jeux a déjà été défini pour l'un de ces systèmes, la Logique linéaire light LLL ([MurawskiOng00,Murawski01]), mais pas pour les autres. Cependant ce modèle ne présente pas de borne de complexité intrinsèque au sens que nous avons évoqué précédemment. Nouveaux langages : de façon plus ambitieuse on peut envisager qu'un modèle de jeux polynomial puisse être un point de départ pour définir de nouveaux systèmes caractérisant P_{time} , peut-être plus flexibles que ceux existants. Il faut observer à ce sujet qu'en logique les sémantiques dénotationnelles ont à plusieurs reprises servi de guide pour définir des logiques; ceci a été le cas déjà pour la logique linéaire qui a été introduite par Girard à partir des espaces cohérents. Dans un esprit un peu différent, mais justement dans le cadre de la complexité, Martin Hofmann a utilisé des modèles de réalisabilité ([Hofmann00,HofmannScott00]) comme guide pour étendre le système de récursion sûre de Bellantoni-Cook caractérisant P_{time} , à l'ordre supérieur.

Objectif intermédiaire: avant d'introduire un modèle de jeux intrinsèquement polynomial, si cette direction se révèle difficile une étape intermédiaire pourra consister à chercher des modèles de jeux spécifiques pour modéliser les systèmes de complexité P_{time} connus. On a déjà mentionné le modèle de LLL, mais les autres systèmes de complexité P_{time} ne disposent pas pour l'instant de sémantiques analogues. Le système LFPL de Hofmann ([Hofmann99]) par exemple, décrit par un système de types permettant de garantir que le calcul est fait en place, semble reposer sur des intuitions qui pourraient être modélisées dans le cadre des jeux.

Visiteurs : Ugo dal Lago.

B. Types, lambda-calcul et réseaux (Patrick Baillot, Virgile Mogbil, [Jean-Batiste Bianquis, Vincent Atassi])

Les logiques linéaires à complexité bornée peuvent être utilisées comme des systèmes de types pour le lambda-calcul, le noyau des langages fonctionnels. Si un terme est bien typé alors, suivant le système de types considéré, il représente une fonction de complexité P_{time} ou élémentaire. Ainsi déterminer si un terme admet un type dans le système est un moyen de garantir statiquement une borne de complexité.

Les problèmes d'inférence de type ont été étudiés pour plusieurs de ces systèmes, comme la Logique linéaire élémentaire (ELL) ou la Logique linéaire light (LLL), et des résultats de décidabilité ont été obtenus ([CoppolaMartini01,CsdR03,Baillot04]). Dans le cas de ELL l'inférence est même faisable en pratique puisqu'on dispose d'un algorithme polynomial ([BaillotTerui05]).

On souhaite poursuivre cette direction de recherche afin d'obtenir un panorama clair de la difficulté de l'inférence de type pour les différents systèmes. Par exemple il reste à déterminer si dans le cas de LLL, de sa variante DLAL ([BaillotTerui04]), ou de la Logique linéaire soft (SLL, [Lafont04]) l'inférence peut être effectuée en temps polynomial. On mettra en pratique ces résultats par des implémentations prototypes qui permettront de les tester sur des exemples (ce qui a déjà été fait dans le cas de ELL).

Outre la garantie de bornes de complexité élémentaire, le typage dans la logique linéaire élémentaire ELL a une deuxième utilité, liée à la compilation des langages fonctionnels. En effet depuis les années 90 on dispose de méthodes de *réduction optimale* pour le lambda-calcul, c'est-à-dire de méthodes d'évaluation qui permettent d'exécuter un programme fonctionnel de manière minimale, sans ajouter de travail redondant. Ceci est réalisé au moyen de techniques de réécriture de graphes et d'une procédure de décision sophistiquée (le tout constituant l'algorithme de Lamping, [Lamping90,GAL92]). L'évaluation est faite de manière minimale, mais le coût de la procédure de décision peut faire perdre le bénéfice de cette stratégie (voir [ACM00]).

Il a cependant été observé que les termes du lambda-calcul typés dans ELL (ou LLL) pouvaient être évalués avec la réduction optimale au moyen d'une procédure très simplifiée (algorithme de Lamping abstrait) ([ACM00]). Ceci a d'ailleurs été une motivation pour étudier les techniques d'inférence de type dans ELL ([CoppolaMartini01], [BaillotTerui05]). L'objectif est d'évaluer les termes en 2 étapes : (i) inférer un type dans ELL, (ii) si le terme est typable appliquer l'algorithme de Lamping abstrait. Nous souhaitons poursuivre cette direction de recherche, tirant parti du fait que nous disposons déjà d'un algorithme efficace pour l'inférence dans ELL. Des questions importantes restent ouvertes comme tout d'abord celle de déterminer une borne de complexité explicite sur l'évaluation par réduction optimale des termes typés dans ELL ou LLL. On souhaiterait aussi pouvoir adapter la méthode dans le cas où le typage échoue, par exemple peut-être en typant certains sous-termes et en les évaluant séparément par l'algorithme de Lamping abstrait.

Enfin une autre direction liée aux systèmes ELL, LLL et au typage concerne l'extension de ces systèmes avec des *types récursifs*. L'observation que les propriétés de ces systèmes restent inchangées par l'ajout de points fixes de types (ou types récursifs) remonte à l'introduction de LLL par Girard ([Girard98]). Cependant cette remarque a été peu exploitée d'un point de vue programmation. Récemment des travaux sur SLL et LLL avec types récursifs ([BaillotMogbil04, DalLagoBaillot04]) ont montré que ceux-ci apportaient des facilités pour la simulation des machines de Turing ou la définition de types de données alternatifs, offrant plus de souplesse.

Il reste cependant à déterminer si l'ajout des types récursifs permet réellement de typer plus de programmes intéressants en pratique. Dans ce cas il serait naturel de chercher à étendre les méthodes d'inférence de type dans ELL ou DLAL mentionnées précédemment aux systèmes avec types récursifs, ce qui semble théoriquement faisable.

Par ailleurs, rejoignant la direction des types récursifs il serait alors aussi intéressant de déterminer si la réduction optimale par l'algorithme de Lamping abstrait est valide pour ces systèmes avec types récursifs, ce qui semble aussi plausible et pourrait élargir le domaine d'utilisation de cet algorithme.

Une autre approche de la complexité dans la logique linéaire a été initiée par Terui et Mairson ([MairsonTerui03, Terui04]) : elle consiste à représenter une fonction non pas par une seule preuve (comme dans LLL) mais par une famille uniforme de preuves, où chacune correspond à une taille de donnée fixée. Dans cette perspective il est suffisant de considérer un petit fragment de la logique linéaire, sans duplication : la logique linéaire multiplicative. La syntaxe la plus appropriée est celle des réseaux de preuves.

Cette notion de représentation est en fait inspirée des circuits booléens, un modèle de calcul classique en théorie de la complexité. Terui a donné des traductions des circuits dans les réseaux de preuves et vice-versa. Ceci a fourni une caractérisation dans les réseaux de preuves des classes de complexité définies à l'aide des circuits (comme NC considérée comme la classe des fonctions effectivement parallélisables). On cherchera à étendre cette approche d'une part à d'autres familles de circuits, d'autre part à d'autres fragments usuels de la logique linéaire.

Visiteurs : K. Terui, S. Martini, P. Coppola.

C. Complexité implicite et expressivité (Patrick Baillot, [Jean-Batiste Bianquis, Ugo Dal Lago, Jean-Yves Moyen, Paulin Jacobé de Naurois])

Dans ce volet du projet nous nous intéressons à étendre les systèmes actuels, voire à les unifier, afin d'obtenir un ou des langages ayant à la fois une large expressivité intensionnelle et une bonne robustesse. Plusieurs voies peuvent être explorées et il faut noter que nous pourrions tirer parti pour cela des résultats des volets A et B.

Une première possibilité consisterait à étendre le système de Hofmann LFPL avec des constructions de la Logique linéaire, notamment une modalité pour la duplication et/ou le polymorphisme. Des travaux récents de Hofmann et Dal Lago sur les modèles de réalisabilité pour LFPL et les logiques linéaires light et soft ([DalLagoHofmann05]) pourront peut-être donner des pistes dans cette direction. Par ailleurs, à la suite du volet sémantique A, un modèle de jeux polynomial pour LFPL serait certainement un bon point de départ pour cette réflexion. D'ailleurs l'approche de Dal Lago et Hofmann repose sur une forme de sémantique puisqu'il s'agit de réalisabilité. Enfin il est possible que les types récursifs se révèlent utiles dans le but de rapprocher la logique linéaire et LFPL, car nous avons montré qu'ajoutés à la logique linéaire soft ils permettaient d'obtenir une meilleure expressivité intensionnelle ([BaillotMogbil04]).

Une deuxième possibilité concerne l'approche des quasi-interprétations et de leur utilisation pour les systèmes de réécriture. Cette approche est plus facile d'utilisation que les autres et a une assez large expressivité intensionnelle. Toutefois elle est limitée à un langage du premier ordre (c'est-à-dire qu'on ne peut pas manipuler de fonctions comme arguments) et le problème de trouver une quasi-interprétation semble en général difficile.

Il serait tentant de chercher à généraliser les quasi-interprétations des systèmes au lambda-calcul, c'est-à-dire un langage d'ordre supérieur. Une première étape pour cela pourrait être de considérer un lambda-calcul typé avec récurseur, comme le système T de Gödel. Si on arrive à traiter cette extension on pourra ensuite s'intéresser à un lambda-calcul avec opérateur de point fixe, comme dans le langage PCF, ce qui requerrait alors d'ajouter des conditions de terminaison (à la manière des ordres de terminaison utilisés au 1^{er} ordre).

Enfin, une troisième direction concerne un objectif un peu différent, mais traitant toujours d'expressivité, dans une autre perspective : il s'agit de considérer l'extension des approches de complexité implicite au cadre du calcul sur les réels, ou sur une structure arbitraire dans le contexte du modèle de Blum-Shub-Smale (BSS) ([BSS89]). Le modèle BSS est un modèle idéalisé dans le sens où il admet pour une structure quelconque des opérations de base à coût unitaire: par exemple l'addition et la multiplication sur les réels. Une motivation pour cela est celle de rendre compte des algorithmes d'analyse numérique ou de mathématiques appliquées avec une notion de complexité correspondant à celle considérée en pratique dans ces communautés. La complexité 'classique' apparaît alors simplement comme un exemple particulier où la structure considérée est $\{0,1\}$.

Bournez, de Naurois et al. ([BCdNM03]) ont montré que certaines approches de complexité implicite pouvaient être adaptées au cadre BSS et caractériser ainsi des classes de complexité sur des structures arbitraires. Leur approche s'est cependant concentrée sur des systèmes à expressivité intensionnelle modeste. A leur suite [BaillotPedicini04] a montré que la Logique linéaire light étendue avec des opérations de base sur une structure permettait de capturer la classe Ptime BSS correspondante. Il semble prometteur de chercher à adapter au cadre BSS des caractérisations de complexité Ptime autorisant l'utilisation de l'ordre supérieur (comme LLL). Ceci fournirait en effet un langage à complexité bornée plus susceptible de représenter des algorithmes d'analyse numérique, où typiquement on doit manipuler des fonctions comme arguments. Du point de vue de la complexité implicite ceci offre un nouveau cadre pour comparer la robustesse des systèmes à complexité bornée. A terme ceci est aussi un domaine prometteur pour comparer l'expressivité intensionnelle des langages en jeu puisqu'il ouvre la possibilité d'examiner un champ nouveau d'algorithmes intéressants.

Visiteurs : J.-Y. Marion, M. Pedicini.

Thèmes	Permanents	Compléments 1 ^{ère} année	Compléments autres années
A. Sémantique	Patrick Baillot Pierre Boudes, Olivier Laurent	Daniel de Carvalho, Ugo Dal Lago	1 post-doctorant
B. Types, lambda-calcul et réseaux	Patrick Baillot, Virgile Mogbil	Jean-Batiste Bianquis, Vincent Atassi	
C. Complexité implicite et expressivité	Patrick Baillot	Jean-Batiste Bianquis, Ugo Dal Lago, Jean-Yves Moyen, Paulin Jacobé de Naurois	1 post-doctorant

4. Contexte international

Les groupes travaillant sur des thèmes voisins sont essentiellement: l'équipe de Hofmann et Schwichtenberg à Munich (notamment avec le projet Mobile Resource Guarantees); le projet Follia en Italie (Turin-Bologne-Rome) et l'équipe d'Abramsky et Ong à Oxford.

Le groupe de Munich a mené des travaux à la fois théoriques (LFPL par exemple) et appliqués (projet MRG). Le projet Follia est orienté essentiellement vers la Logique linéaire et son application à

la programmation fonctionnelle. Le groupe d'Oxford a contribué à l'étude de la Logique linéaire light, ne travaille plus actuellement en complexité implicite mais est un site leader en sémantique des jeux et son application en vérification.

Il faut ajouter à cette liste des petits groupes ou individus très actifs : Neil Jones (Copenhague, Danemark) a mené des travaux majeurs sur la complexité et la terminaison en programmation fonctionnelle; Daniel Leivant (Indiana, US) est une référence en complexité implicite, notamment pour les approches par la récursion; Kazushige Terui (Tokyo, Japon) est un jeune chercheur qui a apporté des contributions importantes dans les approches LL à la complexité. En France il faut noter le groupe de Marion à Nancy, pour les approches par système de réécriture et le groupe de Girard à Marseille pour la Logique linéaire.

Notre projet se distingue par l'objectif d'utiliser la sémantique des jeux comme appui pour guider notre recherche en complexité, ainsi que par la volonté d'unifier différentes approches (alors que les sites mentionnés ont plutôt chacun une approche spécialisée).

5. Originalité par rapport au laboratoire

Par rapport au LIPN : l'équipe LCR du LIPN dispose d'une tradition en Logique linéaire, notamment pour ses liens avec l'intelligence artificielle (planification) et la programmation logique. Ce projet s'appuie sur ce cadre, utilise la logique linéaire dans le thème de la complexité et élargit ce point de vue à la problématique plus large de la complexité implicite. Il lui adjoint de plus l'approche sémantique, peu développée jusqu'à présent au LIPN, mais complémentaire des outils précédents, et qui sera renforcée à la rentrée 2005 par le recrutement comme MCF de Pierre Boudes.

Par rapport à PPS : l'équipe PPS se situe dans la thématique de la sémantique et de la logique, ainsi que de leurs applications à la programmation. Il s'agit d'un site de référence international dans le domaine des sémantiques de jeux, mais qui ne travaille pas actuellement en complexité.

6. Calendrier prévisionnel

1. Première année :

Bien que les principaux participants du thème (A) sont des experts en sémantique et notamment en théorie des jeux, l'originalité de ce thème sur le plan international nécessite un état de l'art précis. Les fortes compétences réunies dans le projet sur ce domaine devraient permettre de développer des modèles cernant nos objectifs.

Le thème (B) s'appuie sur des connaissances déjà acquises par les membres du projet et pour lequel il y a déjà eu des visites, collaborations et publications. Cela explique le fait qu'il y ait déjà des visites planifiées et financées en dehors de ce projet. Cette première année devrait se finir en donnant lieu à publications dans des revues internationales sur les questions, spécifiques à cette thématique, qui nous concernent.

Le thème (C) est renforcé par deux post-doctorants (3 mois université Paris 13 et 1 an CNRS) au LIPN. Il sera donc l'objet d'un fort transfert de connaissances. Nous ne connaissons pas encore la réponse à ce jour concernant un candidat dans cette thématique à poste de demi-ATER.

2. Seconde année :

Le thème Sémantique (A) devrait poursuivre sur l'impulsion de la première année en donnant lieu à des avancées significatives au cours de la seconde année surtout concernant les jeux à complexité polynomiale.

A la lumière des apports de la première année des sous-thèmes de (B), les questions concernant le sous-thème (B4) devraient pouvoir être abordées. Le développement général de (B) devrait par ailleurs profiter des connaissances acquises dans le thème Complexité Implicite (C) car ce dernier en est une motivation.

Les premières avancées sémantiques du thème (A) devraient avoir des retombées directes sur le thème (C) qui pourrait alors se développer dans un cadre unificateur.

Nous envisageons un financement de post-doctorat afin de favoriser l'apport du thème Sémantique (A) au thème (C) : ce point nous semble essentiel à la réussite de notre projet.

3. Troisième année :

Le thème (A) aura donné un cadre sémantique permettant d'envisager une convergence des résultats : les variantes de logiques linéaires à complexité bornées obtenues dans la thématique (C)

pourront être affinées au travers des questions et résultats de la thématique (B) plus "classique" qui traite de la complexité implicite par les systèmes logiques.

Pour conclure ce projet, nous souhaitons organiser une rencontre thématique (1 ou 2 jours) concernant les trois volets de ce projet avec des participants internationaux : les deux dernières (B) et (C) auront vraisemblablement connus des avancées significatives sous un angle différent de celui développé dans ce projet car elles concernent les domaines d'un certain nombre de recherches internationales. Enfin à travers cette rencontre nous souhaitons marquer l'émergence au niveau national comme international de la thématique originale de notre projet : la sémantique de la complexité.

Durant tout le projet sera organisé un groupe de travail hebdomadaire. Pour chaque année, nous prévoyons un stagiaire de Masters 2^{ème} année (ex-DEA) dont les sujets seront à définir en fonction de l'avancement. Chaque étape donnera lieu à la rédaction d'un rapport d'avancement. Nous prévoyons à mi-parcours une journée sur les thématiques du projet lors de visites, ainsi qu'un ou deux jours thématiques pour la clôture du projet.

Bibliographie :

- [ACM00] A. Asperti and P. Coppola and S. Martini. *(Optimal) duplication is not elementary recursive*. Proceedings of POPL '00, pp. 96-107, 2000.
- [AGMO04] S. Abramsky and D. Ghica and L. Ong and A. Murawski. *Applying Game Semantics to Compositional Software Modelling and Verification*. Proceedings of TACAS '04, volume 2988 of LNCS, pp. 421-435, Springer 2004.
- [AJM00] S. Abramsky and R. Jagadeesan and P. Malacaria. *Full Abstraction for PCF*. Information and Computation, 163(2):409-470, december 2000.
- [AbramskyMcCusker99] S. Abramsky and G. McCusker. *Full abstraction for Idealized Algol with Passive Expressions*. Theoretical Computer Science, 227:3-42, september 1999.
- [AspertiRoversi02] A. Asperti and L. Roversi. *Intuitionistic Light Affine Logic*. ACM Transactions on Computational Logic, 3(1):1-39, 2002.
- [Baillot04] P. Baillot. *Type inference for light affine logic via constraints on words*. Theoretical Computer Science, 328(3):289-323, december 2004.
- [BaillotMogbil04] P. Baillot and V. Mogbil. *Soft lambda-calculus: a language for polynomial time computation*. Proceedings of FOSSACS 2004, volume 2987 of LNCS, pp. 27-41. Springer 2004.
- [BaillotPedicini01] P. Baillot and M. Pedicini. *Elementary complexity and geometry of interaction*. Fundamenta Informaticae, 45(1-2), 2001.
- [BaillotPedicini04] P. Baillot and M. Pedicini. *An embedding of the BSS model of computation in light affine lambda-calculus*. Research Report IAC CNR Roma, N.57 (11/2004), november 2004.
- [BaillotTerui04] P. Baillot and K. Terui. *Light types for polynomial time computation in lambda-calculus*. Proceedings of LICS 2004, pp. 266-275, IEEE Computer Society Press, 2004.
- [BaillotTerui05] P. Baillot and K. Terui. *A feasible algorithm for typing in Elementary Affine Logic*. Proceedings of TLCA 2005, volume 3461 of LNCS, pp. 55-70. Springer 2005.
- [BCdNM03] O. Bournez and F. Cucker and P. Jacobé de Naurois and J.-Y. Marion. *Computability over an Arbitrary Structure. Sequential and Parallel Polynomial Time*. Proceedings of FOSSACS 2003, volume 2620 of LNCS, pp. 185-199, Springer 2003.
- [BDER97] P. Baillot and V. Danos and T. Ehrhard and L. Regnier. *Timeless games*. Proceedings of CSL '97, volume 1414 of LNCS. Springer, 1997.
- [BellantoniCook92] S. Bellantoni and S. Cook. *New recursion-theoretic characterization of polytime functions*. Computational Complexity, 2:97-110, 1992.
- [BMM04] G. Bonfante and J.-Y. Marion and J.-Y. Moyen. *Quasi-interpretations: a way to control ressources*. Submitted to Theoretical Computer Science, 2004.
- [BNS00] S. Bellantoni and K. Niggl and H. Schwichtenberg. *Higher type recursion, ramification and polynomial time*. Annals of Pure and Applied Logic, 104(1-3):17-30, 2000.
- [Boudes04] P. Boudes. *Projecting games on hypercoherences*. Proceedings of ICALP '04, volume 3142 of LNCS. Springer, 2004.
- [Boudes05] P. Boudes. *Desequentialization of games and experiments in proof-nets*. Preprint IML 2005-1, 2005.
- [BSS89] L. Blum and M. Shub and S. Smale. *On a theory of computation and complexity over the real numbers: NP-completeness, recursive functions and universal machines*. Bulletin of the American Mathematical Society, 21:1-46, 1989.
- [CoppolaMartini01] P. Coppola and S. Martini. *Typing lambda terms in elementary logic with linear constraints*. Proceedings of TLCA '01, volume 2044 of LNCS, pp. 76-90, Springer 2001.
- [CsdR03] P. Coppola and S. Ronchi della Rocca. *Principal typing in Elementary Affine Logic*. Proceedings of TLCA '03, volume 2701 of LNCS. Springer, 2003.
- [DalLago05] U. Dal Lago. *The geometry of linear higher-order recursion*. Proceedings of LICS 2005, IEEE Computer Society Press, 2005.
- [DalLagoBaillot04] U. Dal Lago and P. Baillot. *On Light Logics, Uniform Encodings and Polynomial Time*. Workshop on Logics for Resources, Processes, and Programs (LRPP'04) (affiliated with LICS 2004 and ICALP 2004), Turku, Finland, July 13, 2004.
- [DalLagoHofmann05] U. Dal Lago and M. Hofmann. *Quantitative models and implicit complexity*. Workshop on Logic and Computational Complety, 2005.
- [Girard87] J.-Y. Girard. *Linear Logic*. Theoretical Computer Science, 50(1-102), 1987.

- [Girard88] J.-Y. Girard. *Geometry of Interaction I: an interpretation of system F*. Proceeding of Logic Colloquium '88. North Holland, 1988.
- [Girard98] J.-Y. Girard. *Light linear logic*. Information and Computation, 143:175-204, 1998.
- [GAL92] G. Gonthier and M. Abadi and J.-J. Levy. *The Geometry of Optimal Reduction*. Proceedings of POPL '92. ACM Press, 1992.
- [GSS92] J.-Y. Girard and A. Scedrov and P. Scott. *Bounded Linear Logic: A modular approach to polynomial time computability*. Theoretical Computer Science, 97:1-66, 1992.
- [Harmer99] R. Harmer. *Games and Full Abstraction for Nondeterministic Languages*. PhD Thesis Imperial College and University of London, 1999.
- [Hofmann99] M. Hofmann. *Linear types and non-size-increasing polynomial time computation*. Proceedings of LICS '99, IEEE Computer Society Press, 1999.
- [Hofmann00] M. Hofmann. *Safe recursion with higher types and BCK-algebra*. Annals of Pure and Applied Logic, 104(1-3):113-166, 2000.
- [HofmannScott00] M. Hofmann and P. Scott. *Realisability models for BLL-like languages*. Workshop on Implicit Computational Complexity, 2000.
- [HylandOng00] M. Hyland and L. Ong. *On Full Abstraction for PCF*. Information and Computation, 163(2):285-408, december 2000.
- [Lafont04] Y. Lafont. *Soft linear logic and polynomial time*. Theoretical Computer Science, 318(1-2):163-180, 2004.
- [Laurent01] O. Laurent. *A Token Machine for Full Geometry of Interaction*. Proceedings of TLCA , volume 2044 of LNCS, pp. 283-297. Springer, may 2001.
- [Laurent04] O. Laurent. *Polarized Games*. Annals of Pure and Applied Logic, 130(1-3):79-123, december 2004.
- [Laird97] J. Laird. *Full abstraction for functional languages with control*. Proceedings of LICS '97, pp. 58-67, IEEE Computer Society Press, 1997.
- [Lamping90] J. Lamping. *An algorithm for optimal lambda calculus reduction*. Proceedings of POPL '90, pp. 16-30, ACM Press 1990.
- [Leivant94] D. Leivant. *Ramified recurrence and computational complexity I: Word recurrence and poly-time*. Feasible Mathematics II, Perspective in Computer Science, pp. 320-343, Birkh\user, 1994.
- [LeivantMarion93] D. Leivant and J.-Y. Marion. *Lambda-calculus characterizations of polytime*. Fundamenta Informaticae, 19:167-184, 1993.
- [MairsonTerui03] H. Mairson and K. Terui. *On the Computational Complexity of Cut-Elimination in Linear Logic*. Proceedings of ICTCS 2003, volume 2841 of LNCS, pp. 23-36. Springer, 2003.
- [MarionMoyen00] J.-Y. Marion and J.-Y. Moyen. *Efficient first order functional interpreter with time bound certification*. Proceedings of LPAR, volume 1955 of LNCS, Springer 2000.
- [MurawskiOng00] A. Murawski and L. Ong. *Discreet games, Light Affine Logic and PTIME computation*. Proceedings of CSL '00, volume 1862 of LNCS, Springer 2000.
- [Murawski01] A. Murawski. *On Semantic and Type-Theoretic Aspects of Polynomial-Time Computability*. PhD Thesis University of Oxford, 2001.
- [Terui04] K. Terui. *Proof Nets and Boolean Circuits*, Proceedings of LICS 2004, pp. 182-191. IEEE Computer Society Press, 2004.

Contrats sur les trois dernières années (effectués et en cours) pour les membres des équipes dont l'implication dans le projet est supérieure à 25%

Nom du membre participant à cette demande	% d'implication	Intitulé de l'appel à projets Source de financement Montant attribué	Titre du projet	Nom du coordinateur
P. Baillot	50%	ACI Sécurité Informatique Montant attribué : 125 000 Euros	CRISS ⁵	R. Amadio
	30%	ACI Nouvelles Interfaces des Mathématiques Montant attribué : 100 000 Euros	GEOCAL ⁶	T. Ehrhard
O. Laurent	80%	ACI Nouvelles Interfaces des Mathématiques Montant attribué : 100 000 Euros	GEOCAL ⁶	T. Ehrhard
V. Mogbil	30%	ACI Sécurité Informatique Montant attribué : 125 000 Euros	CRISS ⁵	R. Amadio
	60%	ACI Nouvelles Interfaces des Mathématiques Montant attribué : 100 000 Euros	GEOCAL ⁶	T. Ehrhard

Demandes de contrats en cours d'évaluation ⁷ pour les membres des équipes dont l'implication dans le projet est supérieure à 25%

Nom du membre participant à cette demande	% d'implication	Intitulé de l'appel à projets Source de financement Montant demandé	Titre du projet
O. Laurent	10%	Projet ANR 'blanc' Montant demandé : 474600 Euros sur 3 ans	Logics for computing and proving

Secteurs disciplinaires

Sciences et technologies de l'information et de la communication (STIC),
 Sciences pour l'ingénieur,
 Chimie,
 Physique,
 Mathématiques et interactions,
 Sciences de l'univers et géo-environnement,
 Sciences agronomiques et écologiques,
 Biologie et santé,
 Sciences humaines et sociales

⁵ CRISS : Contrôle de Ressources et d'Interférence dans les Systèmes Synchrones. Ce projet comporte plusieurs volets. Baillot et Mogbil étaient impliqués essentiellement dans le volet « Programmes fonctionnels d'ordre supérieur » qui était prévu jusqu'à 2005.

⁶ GEOCAL : Géométrie du Calcul

⁷ Veuillez mentionner ici les organismes auprès desquels vous avez déposé un autre projet, en particulier au GIP ANR, que ce soit comme coordinateur ou comme partenaire. Pour chacun donnez le nom de l'appel à projet, le titre du projet et le nom du coordinateur.