

LE PROJET

Exposé scientifique (2 pages maximum) : mettez en valeur le caractère exploratoire du projet, l'originalité des approches et la prise de risque.

Contexte

Logique et calcul. Les liens entre logique et programmation sont particulièrement riches, au point qu'on peut faire précisément correspondre les systèmes logiques de démonstration avec les systèmes de typage des programmes: c'est l'isomorphisme de Curry-Howard. Suite à une analyse précise de cette correspondance, Girard a introduit la logique linéaire [Gir87] sur la base d'une sémantique dite quantitative des programmes fonctionnels [Gir88]: intuitivement, on interprète les termes du lambda-calcul (le paradigme à la base de la programmation fonctionnelle) comme des fonctions analytiques entre espaces vectoriels. Si cette origine explique la «saveur» algébrique de la logique linéaire, il a fallu attendre les travaux d'Ehrhard [Ehr02,Ehr05] pour obtenir une telle sémantique quantitative dans un cadre algébrique standard. Ainsi, on peut voir les programmes fonctionnels comme des combinaisons linéaires (potentiellement infinies) de vecteurs de base, chacun représentant un comportement calculatoire partiel. Par un retour à la syntaxe, ces travaux ont mené à une extension du cadre de la correspondance de Curry-Howard, notamment avec l'introduction de la logique linéaire différentielle [ER06] et du lambda-calcul différentiel [ER03].

En se concentrant sur l'aspect quantitatif cette nouvelle syntaxe, Vaux a défini le lambda-calcul algébrique [Vau09] qui introduit dans le lambda-calcul des combinaisons linéaires de termes. Par des moyens sémantiques, Tasson a caractérisé une propriété de totalité dans ce calcul, qui écarte certains termes problématiques (en particulier la somme nulle de termes qui représente l'échec du calcul) [Tas09]. Par ailleurs, Pagani et Ronchi ont étudié le problème de la solvabilité (la possibilité pour un programme d'interagir effectivement avec le contexte) dans le lambda-calcul différentiel [PR10].

Calcul quantique. L'élaboration d'un langage de programmation pour le calcul quantique est délicate: les briques de base du calcul sont essentiellement des transformations unitaires sur des espaces vectoriels complexes. Dans le cadre de la réécriture, diverses extensions du lambda-calcul ont été proposées, en général en introduisant explicitement des primitives de calcul quantique (voir le survol de Selinger et Valiron [SV10]). Par contraste, Arrighi et Dowek ont proposé une approche originale [AD08], consistant à introduire un lambda-calcul uniquement augmenté de combinaisons linéaires de termes: le lambda-calcul linéaire algébrique. Ce calcul est très proche de celui de Vaux, bien que muni d'une dynamique différente. Valiron a étudié cette dynamique dans un cadre simplement typé qui permet de faire le lien entre les deux formalismes [Val09].

Il s'agit alors d'isoler au sein de ce langage riche les programmes correspondant à des algorithmes quantiques. L'approche est fructueuse, puisqu'Arrighi et Dowek établissent un résultat de non-clonage typique du calcul quantique. Par la suite, Arrighi et Díaz-Caro ont recherché les moyens de restreindre par le typage les programmes considérés pour se rapprocher de l'expressivité de la programmation quantique. Une première étape est de définir un système de types permettant de spécifier les coefficients des combinaisons linéaires qui apparaissent dans les termes [AD09]: le système obtenu admet comme cas particulier une version probabiliste du lambda-calcul.

Concurrence. Dans le domaine de la concurrence, la notion de non-déterminisme est centrale, puisqu'il s'agit d'étudier le comportement de processus dans une situation de partage des ressources. Dans ce domaine, les sémantiques de traces consistent à représenter l'exécution d'un processus par l'ensemble de ses scénarios d'interaction possibles. Beffara a récemment proposé [Bef08] une variante du pi-calcul (le formalisme de référence en programmation concurrente) introduisant des sommes formelles de termes. Celles-ci permettent, au moyen d'une interprétation calculatoire adéquate, de représenter à la fois le choix interne (programme non-déterministe) et le choix externe (environnement non-déterministe). Il développe alors

PEPS Interactions INS2I : APPEL A PROJETS EXPLORATOIRES 2009 - 2010

une sémantique de traces interne au calcul: la forme normale d'un processus est exactement la somme de ses suites d'actions possibles. Cette construction l'a mené à introduire une sémantique de test quantitative [Bef09], qui unifie les notions classiques de *may-* et *must-testing* et les généralise et les rapprochant de modèles effectivement parallèles (dits de *true concurrency*).

Des liens précis entre logique linéaire différentielle et concurrence ont par ailleurs été établis par Ehrhard et Laurent [EL07,EL08], pour lesquels la notion même d'interaction non-déterministe est centrale. Ceci semble indiquer que la logique linéaire différentielle et les concepts associés peuvent constituer un environnement logique adéquat pour l'étude de la concurrence dans un cadre quantitatif.

Objectifs

Les démarches que nous venons de présenter partagent toutes un principe commun: l'introduction de sommes formelles, voire de combinaisons linéaires de termes (preuves, programmes, processus) dans une analyse du calcul qu'on peut qualifier de quantitative. C'est-à-dire qu'au lieu d'étudier les comportements possibles d'un objet comme de simples alternatives, on cherche à quantifier la part de chacun des comportements atomiques qui constituent le comportement global. C'est la portée et la pertinence de ce socle commun que nous souhaitons étudier, en réunissant les acteurs de chacune de ces approches.

Outre les interactions qu'on peut attendre, une tâche essentielle sera de développer la théorie de la réécriture d'ordre supérieur (les propriétés du lambda-calcul) dans un cadre quantitatif: modèles d'exécution, propriétés de séparation, définition de formes normales généralisées (arbres de Böhm), notions de solvabilité, stratégies d'évaluation (appel par nom/appel par valeur), sémantiques dénotationnelles, sémantiques de réalisabilité, *etc.* Les résultats mentionnés plus haut suggèrent une belle richesse de concepts, laissant un grand nombre de questions ouvertes.

Le format de ce projet nous permettra, suivant la qualité et la portée des résultats obtenus, d'évaluer la pertinence d'une collaboration à plus long terme et, le cas échéant, de préciser les contours d'un programme de recherche commun présenté à l'ANR. La répartition des moyens demandés reflète ces objectifs: le financement des déplacements pour des rencontres entre participants et l'organisation d'ateliers de travail constitueront l'essentiel des dépenses.

- [Gir87] J.-Y. Girard: Linear Logic. Theoretical Computer Science 50: 1-102, 1987.
- [Gir88] J.-Y. Girard: Normal functors, power series and lambda-calculus. Ann. Pure Appl. Logic 37, no. 2, 1988
- [Ehr02] T. Ehrhard: On Köthe sequence spaces and linear logic. Math. Struct. Comp. Sci. 12 (5), 2002.
- [Ehr05] T. Ehrhard: Finiteness spaces. Math. Structures Comput. Sci. 15, no. 4, 615-646, 2005.
- [ER03] T. Ehrhard, L. Regnier: The differential lambda-calculus. Theoret. Comput. Sci. 309, no. 1-3, 2003.
- [ER06] T. Ehrhard, L. Regnier: Differential interaction nets. Theor. Comput. Sci. 364, no. 2, 2006.
- [Vau09] L. Vaux: The Algebraic lambda-calculus. Math. Struct. Comp. Sci. 19 (5), 2009
- [Tas09] C. Tasson: Algebraic Totality, towards Completeness. TLCA 2009: 325-340
- [PR10] M. Pagani, S. Ronchi della Rocca: Solvability in Resource Lambda-Calculus. FOSSACS 2010
- [SV10] P. Selinger, B. Valiron: Quantum Lambda Calculus. Book chapter to appear in Semantic Techniques in Quantum Computation, Cambridge University Press, 2010.
- [Val09] B. Valiron: About Typed Algebraic Lambda-Calculi. Manuscript.
- [AD08] P. Arrighi and G. Dowek: Linear-algebraic lambda-calculus: higher-order, encodings and confluence RTA 2008.
- [AD09] P. Arrighi and A. Díaz-Caro: Scalar System F for Linear-Algebraic Lambda-Calculus: Towards a Quantum Physical Logic. In Proceedings of the 6th International Workshop on Quantum Physics and Logic, pp. 206-215, Oxford, UK, 2009.
- [Bef08] E. Beffara, An algebraic process calculus. In Proceedings of the 33rd Annual IEEE Symposium on Logic in Computer Science (LICS 2008), 2008.
- [Bef09] E. Beffara, Quantitative testing semantics for non-interleaving, Technical report, 2009.
- [EL07] T. Ehrhard and O. Laurent, Interpreting a Finitary Pi-Calculus in Differential Interaction Nets. CONCUR 2007.
- [EL08] T. Ehrhard and O. Laurent, Acyclic Solos and Differential Interaction Nets. Submitted, 2008.
- [Kri07] J.-L. Krivine: A call-by-name lambda-calculus machine. Higher Order and Symbolic Computation 20, p.199-207 (2007)