2018-00465 - Post-doctorant (e) : Algebra, Rewriting, and Functional Programming

**Contract type** : Public service fixed-term contract  
**Level of qualifications required** : PhD or equivalent  
**Fonction** : Post-Doctoral Research Visit  
**Corps d’accueil** : Chargé de Recherche (CR)

### About the research centre or Inria department

Located at the heart of the main national research and higher education cluster, member of the Université Paris Saclay, a major actor in the French Investments for the Future Programme (Idex, LabEx, IRT, Equipex) and partner of the main establishments present on the plateau, the centre is particularly active in three major areas: data and knowledge; safety, security and reliability; modelling, simulation and optimisation (with priority given to energy).

The 450 researchers and engineers from Inria and its partners who work in the research center’s 31 teams, the 100 research support staff members, the high-level equipment at their disposal (image walls, high-performance computing clusters, sensor networks), and the privileged relationships with prestigious industrial partners, all make Inria Saclay Ile-de-France a key research centre in the local landscape and one that is oriented towards Europe and the world.

### Context

Functional programming and rewriting theory are intimately related. Namely, the computational model behind functional programming languages is the lambda-calculus, which is the setting where most of the techniques of rewriting theory have been developed before being generalized and made more abstract in order to cover other situations encountered in computer science.

Rewriting theory has also been applied to some branches of mathematics, where it has found many applications to algebra (for instance, the celebrated Knuth-Bendix algorithm was first developed in order to study the theory of groups). While investigating these, deep connections with category theory were found and given rise to higher-dimensional rewriting theory, which is able to deal with higher-dimensional algebraic structures, allowing for the study of fine invariants of those, and show coherence theorems. However, this field is relatively young and many connections between algebra, rewriting and computational models remain to be studied. In particular, linear rewriting, i.e. rewriting on formal sums of terms, is still not well understood in this context, and would have applications to probabilistic and quantum programming languages, as well as in logical foundations of programming such as linear logic.

### Assignment

The first aim of the post-doc is, broadly speaking, to explore and develop topics where algebra, rewriting, and programming meet. We propose two concrete axes of work, but other are possible.

The second aim of the post-doc is to foster interactions between teams of Inria and LIX. The post-doc will indeed work in collaboration with Beniamino Accattoli, from the Parsifal Inria team, and Samuel Mimram, from the Cosynus LIX team. Roughly speaking, Accattoli covers the functional programming part, Mimram the algebraic one, and both have their background in rewriting theory.

### Main activities

We propose two directions of work, one based on tools from linear algebra and one on tools from category theory. Of course, the interaction between algebra, functional programming and rewriting goes well beyond these two topics, and candidates interested in other connections are also welcome.
The first line of work is the rewriting study of the algebraic lambda-calculus, which is an extension of lambda-calculus with linear combinations of programs, and it is at the foundations of probabilistic and quantum functional programming languages. This framework, introduced by Vaux in 2009, has already attracted considerable research, because of its rewriting subtleties, but many questions are still open.

The second line of work is the study of languages allowing the use of monads, e.g. Haskell. We expect that coherence theorems for a lambda-calculus with monads can be achieved through rewriting techniques, from which one can expect to automate the insertion of coercions, made from monad morphisms. This would lead to a practical improvement of state-of-the-art languages (the compiler would automatically insert code, thus simplifying the work of the programmer) which would be theoretically be shown to be sound (the coherence theorem would ensure that no arbitrary decision is taken by the compiler).

Skills
The successful candidate will have a background in linear algebra or category theory. Knowledge of basic rewriting theory is also expected, while expertise in functional programming is less essential.

Benefits package
- Subsidised catering service
- Partially-reimbursed public transport
- Social security
- Paid leave
- Flexible working hours
- Sports facilities

Remuneration
Monthly gross salary: 2653 euros