2018-00500 - PhD Position/ Memory access classification for iterative computing [S]

**Contract type**: Public service fixed-term contract

**Level of qualifications required**: Graduate degree or equivalent

**Function**: PhD Position

**Level of experience**: Recently graduated

---

**Context**

**Research teams**: ICPS, INRIA Nancy Grand Est

**Research Unit**: ICube Laboratory

**Thesis directors**: Jens Gustedt (jens.gustedt@inria.fr), Philippe Clauss (philippe.clauss@inria.fr)

**Location**: Strasbourg/Illkirch, France

**Duration**: 3 years

---

**Assignment**

This PhD thesis project aims the combination of two research streams of our team, namely to integrate models for coarse grained parallelism with (fine-grained) polyhedral models of compilation.

For coarse grained parallelism, with our work on ORWL (ordered read-write locks) [1], we have proposed a framework dedicated to a large set of applications characterized by iterative computations. ORWL guarantees important properties related to equity of all compute tasks and liveness. Currently, we are applying this framework to iterative computations that are decomposable into tasks that are guaranteed to have the same memory access pattern for all iterations. This execution scheme is typical in numerical simulations. Memory access classification is used to detect the data access order of the tasks and to schedule them according an autonomous access pattern.

Currently, the application code is instrumented either manually by means of pragmas (C), or automatically by overloading the access operator [] (C++). With these, the programmer identifies a suitable iteration loop, the decomposable tasks and annotates the memory accesses as being read-only or exclusive. Our framework then instruments the memory access pattern during the first three iterations, sets up a system of FIFOs that regulates the data accesses and then runs the mostly unmodified code for the remaining iterations. The polyhedral model3 is a well-known mathematical framework that describes iterations and memory accesses by means of integral points in polyhedra. State of the art compilers are able to use this framework to automatically reorganize nested iteration loops. Our Apollo platform has been developed to extend the scope of this model from static compilation to runtime speculative parallelization [2, 3]. It allows to instrument and predict memory access patterns dynamically, to choose optimized binary code variants, to execute code speculatively and to roll-back executions that violated the prediction if necessary.

---

**Bibliography**

2. Aravind Sukumaran-Rajam and Philippe Clauss, The Polyhedral Model of Nonlinear Loops, ACM Transactions on Architecture and Code Optimization, Association for Computing Machinery,
Main activities
The goal of this thesis is to formalize our approach of memory access classification in terms of the polyhedral model such that the classification phase can be automated and such that the execution phase can be verified.

For this, it will be crucial to capture regular access patterns of the decomposed tasks as much as possible and to describe them as potentially intersecting polyhedra of the data domain. In addition, it will also be necessary to describe and manage irregular accesses that add to the more regular ones.

This should lead to more competitive execution times for the classification phase, to a verification of the access pattern during execution of the decomposed tasks that can be performed in parallel, and to a rollback strategy that ensures coherent execution if accesses outside the predicted domain occur.

Skills
- good knowledge in parallel computing and algorithms, compilation, polyhedral models
- advanced level of programming, preferable in C, and some knowledge of compiler tools

Benefits package
- Subsidised catering service
- Partially-reimbursed public transport

Remuneration
Monthly salary after taxes: around 1596,05€ for 1st and 2nd year. 1678,99€ for 3rd year. (medical insurance included).

General Information
- **Theme/Domain**: Architecture, Languages and Compilation
- **Scientific computing (BAP E)**
- **Town/city**: Illkirch
- **Inria Center**: CRI Nancy - Grand Est
- **Starting date**: 2018-10-01
- **Duration of contract**: 3 years
- **Deadline to apply**: 2018-05-01

Contacts
- **Inria Team**: CAMUS
- **Recruiter**: Gustedt Jens / jens.gustedt@inria.fr

The keys to success
**Degree**: We expect candidates to have masters degree in computer science or other close domain to the field.
How to apply

Upload your CV on jobs.inria.fr; this should be a pdf file of at most 2MB.

In addition, send the following documents to jens.gustedt@inria.fr in a single ZIP file:

- CV.
- A motivation letter describing your interest in this topic.
- Your degree certificates and transcripts for Bachelor and Master (or the last 5 years if not applicable).
- Master thesis (or equivalent) if it is already completed, or a description of the work in progress, otherwise.
- Publications, if any (it is not expected that you have any)

In addition, at least one recommendation letter from the person who supervises(d) your Master thesis (or research project or internship) should be sent. At most two other recommendation letters may be sent. The recommendation letter(s) should be sent directly by their author to jens.gustedt@inria.fr

Applications to be sent as soon as possible.

Conditions for application

Defence Security:
This position is likely to be situated in a restricted area (ZRR), as defined in Decree No. 2011-1425 relating to the protection of national scientific and technical potential (PPST). Authorisation to enter an area is granted by the director of the unit, following a favourable Ministerial decision, as defined in the decree of 3 July 2012 relating to the PPST. An unfavourable Ministerial decision in respect of a position situated in a ZRR would result in the cancellation of the appointment.

Recruitment Policy:
As part of its diversity policy, all Inria positions are accessible to people with disabilities.

Warning: you must enter your e-mail address in order to save your application to Inria. Applications must be submitted online on the Inria website. Processing of applications sent from other channels is not guaranteed.