Compositional semantics of the reversible $\pi$-calculus
Projet de Stage M2
Jean Krivine, Daniele Varacca
PPS - Université Paris Diderot

Motivation

Programming recoverable systems can be approached by adopting a radical position. Recognizing that all system recovery schemes rely on some form of undo or rollback, we would like to study the following question: what if we could UNDO any single action in a distributed program?

By adopting a reversible model of computation, one can develop systematic and composable abstractions for recoverable and dependable programming. Besides natively supporting various fault-recovery schemes, a reversible model of computation offers several potentially important benefits: it naturally supports a very general notion of communicating transaction; it provides a sound basis for distributed debugging, fault diagnosis and analysis; it natively supports speculative execution and simulation of distributed programs.

However, developing a reversible distributed programming model raises several issues. In particular, what does it mean to reverse a distributed computation? In which order should it be undone? One approach is to reverse the actions precisely in the reverse order in which they happened. But the exact order of actions in a distributed computation is not important, and we may accept to reverse them in a different order, as long as at least the causality between actions is respected. But which notion of causality shall we consider then?

In the context of the ANR project rever which will start in January 2012, we propose a 4 month internship at PPS laboratory in order to study preliminary issues pertaining to reversibility and causality in the context of the $\pi$-calculus [Mil99], a fundamental model of distributed computation.

Context

Reversible computing in a broad sense has already a long history, which originated with motivations related to physics, notably the observation that only irreversible operations in a digital computer necessarily dissipate heat and that the development of future energy-efficient computing devices could require exploiting reversibility. These original motivations led to the development of theoretical models of computation, as well as to more recent developments on reversible sequential programming languages [THR08], on models of biochemical processes [DK03], on proposals for quantum programming languages [AG05].

More directly relevant to this internship, there are recent works which study the introduction of reversibility in concurrent languages, notably in CCS [Kri06,
DK04, DK05], in process calculi defined by means of SOS rules in GSOS format [PU07], and in the (higher-order) \( \pi \)-calculus [LMS10]. As hinted above, and also noted in [PU07], in concurrent models, the notion of reversibility is tightly connected to the notion of causality. Recent work establishing causal models for the pi-calculus are [VY10, CVY11].

Objectives

**Reversible transition system** The main aim of the internship will be to propose a labelled reversible transition system for the \( \pi \)-calculus, based on the idea from [Kri06, CVY11]. To validate the proposal it will be important to provide some correctness results, relating the reversible semantics to either the event structure or the standard semantics.

**Configuration structures** If time permits the students will begin the study of the properties of the configurations structures [vGP09] that correspond to the reversible semantics.

**Equivalences** It can also be interesting to study the notion of causal/reversible equivalences induced by the proposed model.

Research Environment

The internship, that will last 20 weeks, will be co-directed by Jean Krivine and Daniele Varacca at the laboratory PPS, University Paris Diderot. A stipend of 417.09 euros per month will be provided according to the University regulation.

Expected profile

The internship is addressed to M2 students with a background in theoretical computer science. Some knowledge in concurrency theory would be a plus.

Références


