Non-Volatile Computability

Goal: Characterize the computing power of non-volatile memory models.

Tools: Logic, algorithmic reasoning, programming

Prerequisites: basic knowledge of distributed algorithms, basic concurrent programming skills, curiosity and persistence

Summary

Concurrent programs are often expected to provide safety and progress in asynchronous systems in the presence of crash failures [7]: a faulty process prematurely stops taking steps of its algorithm. Recently, a lot of attention has been paid recently to crash-recovery models [2] in which a process can resurrected after a crash. This was driven by the emerging non-volatile memory in which main conventional memory is equipped with the persistence feature. Recoverable objects designed for such models allow their operations to recover from crash failure [1,5,6].

The new model forces us to reconsider classical distributed computability results [3, 4, 7, 8], separating computable from not computable for a given model of computation. The goal of this project is to determine the computability bounds in the non-volatile context.

Contact

Prof. Petr Kuznetsov
http://www.infres.enst.fr/~kuznetso/
petr.kuznetsov@telecom-paristech.fr
INFRES, Télécom ParisTech
Office C213-2, 46 Rue Barrault

References


