

Continuations, closed reduction and process networks

Ian Mackie

Abstract

In the early 1990s it was shown that the linear λ -calculus (the usual λ -calculus restricted so that copying and discarding are not allowed) is an internal language for symmetric monoidal closed categories. With the introduction of a natural numbers object—represented in the internal language with numbers and an iterator—it was shown, using a result of Robert Paré and Leopoldo Román, that all primitive recursive functions could be represented.

Around the same time interaction nets were introduced as a simple graph rewriting system that generalise proof nets of linear logic. Rules have a linearity constraint, and as a consequence, explicit copying and discarding are required. One area of application of interaction nets is the representation of the λ -calculus, where many different systems have been developed.

Kahn process networks (1970s) are a model of computation based on a collection of sequential, deterministic processes that communicate through unbounded channels. They are well suited for modelling stream-based computations, but are in no way restricted to this application.

In this talk we connect the above by giving an interaction net encoding of process networks. We give just one part of this story and focus on the linear λ -calculus with iterators. As part of the compilation, we require an understanding of closed reduction, and this in turn will make use of continuations.