Understanding functional self-application

Reinhard Kahle

kahle@mat.uc.pt
Centria, Universidade Nova de Lisboa and
Departamento de Matemática, Universidade de Coimbra,
Apartado 3008, P-3001-454 Coimbra, Portugal

Abstract

While functional self-application, i.e., the possibility of a function to be its own argument, is usually not considered in mathematics, it turned out that this concept can be meaningfully integrated in logical systems and programming languages. In mathematical logic it was first studied in the context of *combinatory logic* introduced by SCHÖNFINKEL in 1920 and later on developed by CURRY and by CHURCH in the twin theory of *λ calculus*. It is an essential feature of type-free functional programming languages, but also of logical frameworks like FEFERMAN's systems of *explicit mathematics*. Moreover, from the theoretical point of view, self-application can be used to prove standard *undefinability results*.

In this talk we show how functional self-application can be considered as a special case of *diagonalization*. In this perspective, it looses all its mystery; in particular, a *recursion-theoretic* reading of functional self-application allows to understand it as well-known concept, used not only in logic, but implicitly also in every modern computer.