A coinductive approach to digital computation

Ulrich Berger
Swansea University, United Kingdom

20th July 2008

Abstract

We present digit systems and their coinductively defined morphisms as an abstract approach to digital computation. A digit system is a set X together with a set D of functions, from X to X. The elements of D are called digits. A standard example of a digit system is the compact real interval $I = [-1, 1]$ together with the functions $\mathrm{av}_i : I \to I$ $(i = -1, 0, 1)$ defined by $\mathrm{av}_i(x) = (x + i)/2$. This digit system corresponds to the well-known binary signed digit representation of real numbers in $[-1, 1]$ which has been extensively studied in exact real number computation, type theory and domain theory.

Our approach contains two novelties:

1. We define coinductively a set of morphisms between digit systems which, in the standard cases, coincides with the set of uniformly continuous functions.

2. We use the proof-theoretic technique of program extraction to automatically synthesise from constructive proofs lazy algorithms for these morphisms.

We show how constructive analysis and corresponding certified algorithms can be developed in this approach.

Structures similar to digit systems are known as iterated function systems in the theory of dynamical systems and fractals. Since our theory has different goals we chose a different name.

We also define the domain-theoretic semantics of the data structures involved in this approach and discuss questions of computability and termination.