Interaction morphisms and Turing computation

Tarmo Uustalu
Tallinn University of Technology
Tallinn, Estonia

I will introduce interaction morphisms as a means to specify how an effectful (e.g., non-deterministic, interactive I/O or stateful) computation is to be run on an abstract state machine. An interaction morphism is given by a monad $T = (T, \eta, \mu)$ and a comonad $D = (D, \varepsilon, \delta)$ on a Cartesian category with a family of maps $\psi_{X,Y} : TX \times DY \to X \times Y$ natural in X and Y and agreeing suitably with η, ε, μ, δ. Intuititively, $\psi_{X,Y}$ takes a computation and a behavior from an initial state and sends them into a return value and a final state. Interaction morphisms enjoy neat properties: they are the same as monoids in a certain monoidal category; interaction morphisms of T and D are in a bijective correspondence with carrier-preserving functors between the categories of coalgebras of D and stateful runners of T (monad morphisms from T to state monads); they are also in a bijective correspondence with monad morphisms from T to a monad induced in a certain way by D.

I will illustrate interaction morphisms on the example of Turing computation, i.e., computations interacting with a readable, writeable, walkable biinfinite tape storing symbols from a finite alphabet. Describing the monad and comonad concretely is an instructive exercise in this case.

The work on interaction morphisms continues my earlier work [2] on stateful runners and is joint with Shin-ya Katsumata (Kyoto University). Turing computation was studied in a related setting by Goncharov et al. [1].

References

[1] S. Goncharov, S. Milius, A. Silva. Towards a coalgebraic Chomsky hierarchy. TCS 2014, v. 8705 of Lect. Notes in Comput. Sci., pp. 265–280, Springer, 2014.

[2] T. Uustalu. Stateful runners for effectful computations. MFPS XXXI, v. 319 of Electron. Notes in Theor. Comput. Sci., pp. 403–421, Elsevier, 2015.