Convergence Classes and Spaces of Partial Functions

Pascal Hitzler*

Artificial Intelligence Institute
Dresden University of Technology
Dresden
Germany
phitzler@inf.tu-dresden.de
www.wv.inf.tu-dresden.de/∼pascal/

Anthony Karel Seda

Department of Mathematics
University College Cork
Cork
Ireland
a.seda@ucc.ie
maths.ucc.ie/staff/seda/

We consider the use of convergence classes of nets, as expounded in Kelley's book "General Topology", for example, as a means of generating topologies of interest in computation. We view such classes as a versatile way of introducing suitable topologies on spaces of truth values and interpretations, and on domains $(X \to Y)$ of partial functions from a set X to a set Y. In particular, we illustrate this point of view by analysing the domain-theoretic content of the form of the Vienna Development Method, VDM$^{\clubsuit}$, developed in Trinity College Dublin by Mac an Airchinnigh and his collaborators.

The Vienna Development Method (VDM) of formal specification of software systems was originally developed by IBM at its research laboratories in Vienna with the aim of specifying the formal semantics of programming languages. Over time, VDM has evolved into much more than simply a means of formal specification and has become a powerful development method starting with the formal specification of the system requirements and ending, after a sequence of refinement steps, with the implemented program code.

Where VDM$^{\clubsuit}$ differs from VDM is that, whilst employing preconditions, it does not employ postconditions and the consequent demonstration of their satisfaction by use of formal logic. Instead, operations are explicitly constructed and proof obligation (of system invariant) is carried out mathematically. Therefore, the emphasis in VDM$^{\clubsuit}$ is on constructive mathematics in contrast to the use of formal logic, and, in order to facilitate such proof, an extensive calculus has been developed in VDM$^{\clubsuit}$ to handle a number of operators defined on spaces of partial functions. These operators include, amongst others, domain and range removal, extension of functions and in particular override of functions (which is an important tool in modelling the process of updating records, file systems etc.).

Our objective here is to consider these operators from the point of view of, say, LCF and to determine which of them are Scott continuous and which are not. In fact, some of the more important operators, such as the override, are not Scott continuous, and we consider alternatives to the Scott topology to handle those operators. It turns out that convergence classes give an elementary, but quite pretty, solution to these problems.

*The first named author acknowledges partial support under grant SC/98/621 from Enterprise Ireland.