The Category-theoretic Solution of Recursive Ultra-metric Space Equations

Amin Timany
iMinds-DistriNet – KU Leuven
firstname.lastname@cs.kuleuven.be

Bart Jacobs
iMinds-DistriNet – KU Leuven
firstname.lastname@cs.kuleuven.be

Abstract

We give a short description of our implementation in Coq supporting the construction of category-theoretic solutions to recursive ultra-metric space equations for domain theory. This is one step in our efforts to provide a category-theoretical foundation for program semantics and program logics.

Introduction One particular difficulty in defining a denotational semantics for concurrent higher-order imperative programming languages is the fact that their models are solutions to recursive and sometimes circular equations. The same applies to defining semantics of program logics for verification of programs written in these languages. This is because a programming language's semantics is usually used to reason about soundness of such program logics. To be more specific, to interpret an imperative programming language with higher-order store we need a model of the program's heap $\mathcal{W} = \mathbb{N} \rightharpoonup_{\text{fin}} \mathcal{T}$ (usually referred to as worlds in the Kripke style semantics) that maps memory locations to types. Here, finiteness is to indicate that only a finite number of memory locations are allocated. On the other hand, to interpret types, we need to interpret reference types ("ref τ" for a type τ) depending on the world (state of the heap) at hand. That is, ref τ for a world w should be interpreted as the set of memory locations that have type τ in w. Hence, types should be interpreted as $\mathcal{T} = \mathcal{W} \rightarrow_{\text{mon}} 2^V$ where V is the set of values (including memory locations). The monotonicity requirement is for the sake of coherence. That is, allocating more memory locations should not shrink the set of values of a type. In case of reference types it should indeed increase them. This evident circularity is the main cause of difficulties faced in defining semantics of such programming languages. This is best explained in [2].

Solution in M-categories Such recursive and circular domain-theoretic equations are usually solved in a category enriched over a category with extra structure which allows construction of such solutions. These solutions are usually in the form of the fixed points of some functors unique up to isomorphism. Such a method is presented in [1] and compared to some other relevant works.

In [1], the extra structure is that of a non-empty complete bounded ultra-metric space. An ultra-metric space consists of a set M and a distance function $\delta : M \times M \to \mathbb{R}^+$ to the positive real numbers such that:

UM-1 $\forall x, y.\ \delta(x, y) = 0 \Leftrightarrow x = y$
UM-2 $\forall x, y.\ \delta(x, y) = \delta(y, x)$
UM-3 $\forall x, y, z.\ \delta(x, y) \leq max(\delta(x, z), \delta(y, z))$

An ultra-metric space is *complete* if every Cauchy sequence has a limit. It is *bounded* if the codomain of δ instead of $\mathbb{R}^+$ is the set $[0, b]$ for some $b \in \mathbb{R}^+$. A function $f : M \to M'$ from one ultra metric space to another is called *non-expansive* if $\delta'(f(x), f(y)) \leq \delta(x, y)$ and *contractive* if there is a $c < 1$ such that $\delta'(f(x), f(y)) \leq c \cdot \delta(x, y)$.

Intuitively, the distance function of an ultra metric space can be thought of as the degree of similarity of two elements rather than their spatial distance. One particular class of ultra-metric spaces are bisected ultra-metric spaces where the distance function $\delta : M \times M \to \{0\} \cup \{2^{-n} | n \in \mathbb{N}\}$. As an instance, consider the space of functions from natural numbers to a set A where the distance is defined as:

$$\delta(f, g) = \begin{cases} 0 & \text{if } f = g \\ 2^{max\{n | \forall m<n.\ f(m)=g(m)\}} & \text{otherwise} \end{cases}$$

In [1], the authors call a category $\mathcal{C}$ an M-category if it is enriched over the category of non-empty complete 1-bounded ultra-metric spaces $CBULT_{ne}$. It is furthermore required that the composition operation for morphisms of $\mathcal{C}$ forms a non-expansive function. The domain of this composition function is taken to be the product of the ultra-metric spaces in the usual sense (the product in $CBULT_{ne}$).

A functor $\mathcal{F} : \mathcal{C} \to \mathcal{D}$ from one M-category to another is called *locally non-expansive* and *locally contractive* if its morphism maps are respectively non-expansive and contractive. In [1], the authors show that any mixed-variance locally-contractive functor $\mathcal{F} : \mathcal{C}^{op} \times \mathcal{C} \to \mathcal{C}$ has a unique fixed point up to isomorphism whenever $\mathcal{C}$ has a terminal object and limits (category theoretical notion of limit) of some class of functors which they call increasing Cauchy towers. An object A is a fixed point of $\mathcal{F}$ if we have $\mathcal{F}(A, A) \simeq A$.

Implementation We have implemented (see [5]) the theory of construction of solutions to locally-contractive functors of [1] in Coq. This implementation is based on our general implementation of category theory [4]. In what follows we describe this development and give a brief comparison between this work and the other very recent independent development implementing this theory [3].

Ultra-metric spaces In order to avoid working with real numbers and having to deal with their idiosyncrasies, we have developed a more general notion of ultra-metric spaces. That is, instead of real numbers we use what we call an M-lattice. An M-lattice is a preorder relation $(X, \sqsubseteq)$ such that

ML-1 Has a bottom element $\perp$
ML-2 Has meet ($\sqcup$) of arbitrary subsets of X
ML-3 Has a top element $\top$
ML-4 $Appr(X)$ is a subset of X of approximation elements
ML-5 $\forall a.\ a \in Appr(X) \to \perp \sqsubset a$
ML-6 $\forall a.\ \perp \sqsubset a \to \exists b \in Appr(X).\ b \sqsubseteq a$
ML-7 $\forall a.\ (\forall b \in Appr(X).\ a \sqsubset b) \to a = \perp$
ML-8 $(\forall a \in Appr(X).\ \exists b \in Appr(X).\ b \sqsubset a) \vee$
$(\exists c \in Appr(X).\ \forall a.\ a \sqsubset c \to a = \perp)$

Conditions **ML-1** and **ML-2** imply that X is a complete meet-lattice (in the order-theoretic sense). The elements $\perp$ and $\top$ respectively play the role of 0 and b (the bound).

In practice we only care about approximations (e.g., of limits) only for distances in $Appr(X)$. For instance, one can work with real numbers but only care for approximation of limits only up to rational numbers. The rest of the conditions are to allow us to prove that defining limits with approximations up to approximation elements have the desired properties, e.g., uniqueness and Banach's fixed point theorem. The disjunction and existential quantifiers in Condition **ML-8** are respectively represented as sum types and Σ types (dependent sum type) to allow their elimination in computational contexts.

The notion of M-lattice as described above allows us to develop a general theory of ultra-metric spaces. They allow us to prove general properties required, e.g., the fact that $CBULT_{ne}$ itself forms a complete cartesian-closed M-category. In [3], the authors only provide support for bisected ultra-metric spaces. We represent bisected spaces by providing an M-lattice whose elements are monotone functions (in Coq) $f : \mathbb{N} \to$ `Prop`. By monotone we mean $\forall n, m.\ m \leq n \to f(n) \to f(m)$. In this encoding $\top$ is (`fun n ⇒ False`) and $\perp$ is (`fun n ⇒ True`) and $f \sqsubseteq g$ iff $\forall n.\ g(n) \to f(n)$ (note the order of implication). The distance of two elements of a bisected ultra-metric space is then defined as $\delta(x, y)(n) = \bigwedge_{0 \leq i \leq n} x(n) = y(n)$ which is obviously monotone.

Contractiveness In order to represent contractiveness without a contraction factor c, we use a function called contraction rate $\rho : L \to L$ where L is the M-lattice representing distances. It is required that ρ is monotone, non-expansive ($\forall x.\ \rho(x) \sqsubseteq x$), contractive for positive distances ($\forall x.\ \perp \sqsubset x \to \rho(x) \sqsubset x$) and that $\forall x, y \in Appr(L).\ \exists n \in \mathbb{N}.\ \rho^n(x) \sqsubset y$. Intuitively and practically in our definition of contractive functions, $\rho(x)$ plays the role of $(c \cdot x)$. In [3], working with bisected spaces, the authors require a contractive function f to satisfy the rather strong condition that $\delta(f^n(x), f^n(y)) \leq 2^{-n}$.

Equality In our definition of an ultra-metric space in Coq we have used the conditions of ultra-metric spaces specified above verbatim. For equality, we have used Coq's internal definition of equality. This is contrary to [3]. There, the authors use setoids to have custom equalities. In particular, they want to say two elements are equal if their distance is less than 2^{-n} for any $n \in \mathbb{N}$. We prove this using axioms of functional and propositional extensionality.

M-categories In our development, M-categories are simply categories where the morphism sets form a complete bounded ultra-metric space. Note that we don't require the rather restrictive condition of non-emptiness. To compensate for this, we require the user to provide a morphism $f : 1 \to \mathcal{F}(1, 1)$ for construction and proof of uniqueness of the fixed point of $\mathcal{F}$. Here 1 is the terminal object of the category. This is the only place where the non-emptiness is actually used in the construction and proofs. The authors of [3] drop the non-emptiness condition just as we have done.

Categories in [4] are represented using records. We define M-categories as a record as follows:

```
Record MCat (L : MLattice) : Type :=
  { MC_Obj : Type;
    MC_Hom : MC_Obj → MC_Obj → (Complete_UltraMetric L);
    MC_compose : forall {a b c : MC_Obj}, NonExpansive
          (product_CUM (MC_Hom a b) (MC_Hom b c)) (MC_Hom a c);
     ...
    MC_Cat :> Category := {| Obj := MC_Obj; Hom := MC_Hom;
        compose := fun _ _ _ x y ⇒ MC_compose (x, y);
        ... |} }.
```

where `product_CUM` is the product of complete bounded ultra-metric spaces. `MC_Cat` is a let-in projection of the `MCat` record and provides a coercion to the type of categories. This means that any M-category can be simply used as a category whenever necessary.

Moreover, given the η rule for records new in Coq 8.5, whenever we create an M-category $\mathcal{C}_m$ from an existing category $\mathcal{C}$ by showing that $\mathcal{C}$'s composition is non-expansive, the underlying category of $\mathcal{C}_m$ produced by the `MC_Cat` is definitionally equal to $\mathcal{C}$. By using definitions similar to `MCat` above, we get definitional equalities for underlying functors of locally non-expansive and locally contractive functors as well. This means that we have all the definitions, lemmas, etc. provided in [4] at our disposal to use with M-categories and their functors. The authors of [3] have not based their development on any general purpose category theory development and define everything, e.g., functors, (co)limits etc., from scratch and specific to M-Categories.

Fixed points We prove the existence of fixed points of mixed variance locally contractive functors the same way as in [1]. The proof of uniqueness of fixed points in our development is slightly different. This is due to our different requirements, i.e., lack of non-emptiness condition. Therefore, we prove that the fixed point of $\mathcal{F} : \mathcal{C}^{op} \times \mathcal{C} \to \mathcal{C}$ constructed is unique in the full subcategory of $\mathcal{C}$ where every object A has a morphism $f_A : 1 \to A$. This subcategory is indeed an M-category in which any morphism set is non-empty. We show that the fixed point constructed is in this subcategory and furthermore that any two fixed points of $\mathcal{F}$ in this subcategory are isomorphic. The implementation of [3] does not provide any proof of uniqueness. We use all the necessary category-theoretical concepts and lemmas, e.g., (co)limits, their uniqueness up to isomorphism, etc., that are necessary to prove existence and uniqueness of the fixed points from [4].

Use of axioms Our use of axioms is not limited to the axioms of propositional and functional extensionality mentioned earlier. We have also had to made use of some other axioms. Namely, we have used the axiom of constructive indefinite description on the natural numbers and `not_all_ex_not` from the library of Coq to prove Condition **ML-6** for bisected spaces. This is in turn only used to show that whenever two sequence have their pointwise distances less than some $\perp \sqsubset a$ then their limits have a distance of at most a. We use the latter lemma together with excluded middle on whether some distance is the $\perp$ distance or not to prove that some function is non-expansive which is required for the proof that $CBULT$ (we don't construct $CBULT_{ne}$) has all exponentials. In [3], by using setoids and considering only bisected spaces the authors require no axioms to prove similar results.

References

[1] L. Birkedal, K. Støvring, and J. Thamsborg. The category-theoretic solution of recursive metric-space equations. *Theoretical Computer Science*, pages 4102–4122, oct 2010. .

[2] L. Birkedal, B. Reus, J. Schwinghammer, K. Støvring, J. Thamsborg, and H. Yang. Step-indexed kripke models over recursive worlds. In *Proceedings of the 38th Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, POPL '11, pages 119–132, New York, NY, USA, 2011. ACM. .

[3] F. Sieczkowski, A. Bizjak, and L. Birkedal. ModuRes: A Coq Library for Modular Reasoning About Concurrent Higher-Order Imperative Programming Languages. In C. Urban and X. Zhang, editors, *Interactive Theorem Proving*, volume 9236 of *Lecture Notes in Computer Science*, pages 375–390. Springer International Publishing, 2015. .

[4] A. Timany. Categories, 2015. URL https://github.com/amintimany/Categories.

[5] A. Timany. Category-theoretical Solution of Recursive Metric-Space Equations, 2015. URL https://github.com/amintimany/CTDT.