Product Operations in Strong Monads

Reinhold Heckmann

FB 14 – Informatik
Universität des Saarlandes
Postfach 151150
D-66041 Saarbrücken, Germany
e-mail: heckmann@cs.uni-sb.de

September 30, 1997

Abstract

If a strong monad M is used to define the denotational semantics of a functional language with computations, a product operation $\bar{\times} : MX \times MY \to M(X \times Y)$ is needed to define the semantics of pairing. Every strong monad is equipped with two standard products, which correspond to left-to-right and right-to-left evaluation. We study the algebraic properties of these standard products in general. Then we define alternative products with similar properties for strict and parallel evaluation in the special case of strong monads in $\mathcal{DCPO}$ which are obtained as free constructions w.r.t. various theories of non-deterministic computation, including both classical and probabilistic theories.

1 Introduction

Following the proposals of Moggi [12, 13], functional languages with various notions of *computations* can be denotationally described by means of (particular kinds of) *monads*. Informally, these monads are constructions mapping domains of values into domains of computations for these values. Computations involving non-deterministic choice, for instance, are handled by *power domain constructions* [14, 15, 1, 7, 8].

In Moggi's work, the semantics of pairing (A,B) is handled in a standard way that corresponds to left-to-right evaluation. Of course, a dual treatment corresponding to right-to-left evaluation is easily available. In this paper, we investigate other kinds of pair evaluation, e.g., non-deterministic choice of the evaluation order, or parallel evaluation of the components. We propose that the monads that Moggi uses in his semantic descriptions be extended by a semantic operation which models the evaluation of pairs. This alternative product will then override the standard product, and thus give rise to alternative kinds of evaluation different from the standard left-to-right evaluation.

The paper is organized as follows: In Section 2, we define *strong monads* in Cartesian closed categories. In Section 3, we study the two standard products '$\overrightarrow{\times}$' (left-to-right evaluation) and '$\overleftarrow{\times}$' (right-to-left evaluation), which exist for every strong monad. Their algebraic properties should be enjoyed by every alternative product. In Section 4, we concentrate on power domain constructions as special kinds of strong monad. The algebraic properties of their power domains can be described by non-deterministic theories. There are three main examples: the set-like theory S, where choice is commutative, associative, and idempotent; the multiset-like theory M without idempotence; and the probabilistic theory P, where alternatives are chosen with equal probability. In Section 5, we show how strong monads can be obtained as free constructions w.r.t. these algebraic theories. Section 6 is devoted to the specific properties of the standard products in these free power constructions. Then we introduce our first alternative product in Section 7, which non-deterministically chooses the evaluation order. In Section 8, two other alternative products are introduced axiomatically: the strict product and the parallel product. Their main properties are derived from their axiomatic definition. Both are uniquely determined if they exist. In Section 9, we show the existence of parallel products for a large class of theories. The existence of strict products can be shown analogously.

2 Monads and Strong Monads

We define monads according to Definition 1.2 in [12]. There is a different definition in [10], but the two definitions are equivalent as pointed out in [12].

Definition 2.1

A *monad* in a category C is a triple (M, r, E) where

- M maps objects of C into objects of C,
- r is a family of arrows $\mathsf{r}_X : X \to MX$ for every object X of C,
- For every arrow $f : X \to MY$, there is an arrow $Ef : MX \to MY$ such that

(E1) $Ef \circ \mathsf{r}_X = f \quad \text{for } f : X \to MY$,

(E2) $E\,\mathsf{r}_X = \mathsf{id}_{MX}$,

(E3) $E\,g \circ Ef = E\,(E\,g \circ f) \quad \text{for } f : X \to MY \text{ and } g : Y \to MZ$.

We shall often drop the type subscripts of r_X.

According to Moggi's intuitions, the construction M maps domains of values of some type into domains of computations of this type; r maps values v into a trivial computation $\mathsf{r}v$ that immediately *results* in v; and E *extends* computation-valued functions defined on values to those defined on computations. An extended function Ef applied to some computation will execute this computation and apply f to the resulting value(s), which in general involves some more computation.

Defining $Mf = E(\mathsf{r}_Y \circ f)$ for $f : X \to Y$ makes M into a functor; r then becomes a natural transformation.

An example for a monad in the category $\mathcal{SET}$ is provided by the powerset construction. It has $MX = \{A \mid A \subseteq X\}$, $\mathsf{r}x = \{x\}$, and $EfA = \bigcup_{a \in A} fa$ for $f : X \to MY$ and A in MX. The reader is invited to check the validity of the monad axioms. The induced action of M on functions is given by $MfA = E(\mathsf{r} \circ f)A = \bigcup_{a \in A}\{fa\} = \{fa \mid a \in A\}$, which is the image of A under f.

In [12], a *strong monad* in a Cartesian category is defined as $(M, \mathsf{r}, E, \mathsf{t})$, where $\mathsf{t}_{XY} : X \times MY \to M(X \times Y)$ is an additional operation subject to five axioms. This definition suggests some degree of freedom in extending a monad to a strong monad, which actually does not always exist: as shown in [12], t is — once it exists — uniquely determined by its axioms if the underlying category is concrete.

The situation becomes simpler if we restrict ourselves to the case of Cartesian closed categories [10]. As also mentioned in [12], a strong monad is then given by a triple $(M, \mathsf{r}, \mathsf{E})$, where E is no longer a (set-theoretic) function between hom-sets, but an arrow $\mathsf{E} : [X \to MY] \to [MX \to MY]$ between function objects. The monad axioms have then to be understood as written in the *internal language* [9] of the Cartesian closed category. This internal language is a variant of a simply typed λ-calculus. Axiom (E1) for instance can still — though a bit informally — be written as "$\mathsf{E}f \circ \mathsf{r} = f$ for $f\colon X \to MY$", which stands for the equality statement $f\colon X \to MY \vdash \lambda x^X.\, \mathsf{E}f(\mathsf{r}x) = f$. *From now on, all equations actually mean equality statements of the internal language, written slightly informally.*

As we have derived M from E above, we can derive an arrow M from E in case of a strong monad by defining $\mathsf{M}f = \mathsf{E}(\mathsf{r} \circ f)$. (This definition is to be considered syntactic sugar for the actual definition $\mathsf{M} = \lambda f^{X \to Y}.\, \mathsf{E}(\lambda x^X.\, \mathsf{r}_Y(fx))$ in the internal language.)

In the category $\mathcal{SET}$, every set-theoretic function is an arrow, whence every monad is strong. In the category $\mathcal{DCPO}$, a monad (M, r, E) is strong iff the (set-theoretic) function E is Scott continuous.

3 The Two Standard Products of a Strong Monad

Given a strong monad $(M, \mathsf{r}, \mathsf{E})$ in a CCC C, one can translate expressions of a typed functional language with computation into the internal language of C, once the base types β of the language have been translated into objects β^* of C. This translation is extended to all types τ in a standard way. The domain of semantic values of type τ is then τ^*, and the domain of computations of type τ is $M\tau^*$.

Every expression $E : \tau$ of the language denotes a computation of type τ. This computation is obtained by a syntactic translation of the expression E into a λ-expression $[\![E]\!] : M\tau^*$. In this paper, we are only interested in one aspect of this: the translation of pairs. Without committing ourselves to anything, we may define $[\![(A, B)]\!] = [\![A]\!] \mathbin{\bar{\times}} [\![B]\!]$, where $\bar{\times}$ is an operation which — according to the typings involved — must have type $MX \times MY \to M(X \times Y)$. (More accurately, $\bar{\times}$ is a family of operations $\bar{\times}_{XY}$, but we drop the type subscripts.)

After putting it into our notation, Moggi's choice for '$\bar{\times}$' becomes

$$A \mathbin{\vec{\times}} B = \mathsf{E}(\lambda a.\, \mathsf{E}(\lambda b.\, \mathsf{r}(a, b))B)A,$$

which corresponds to a left-to-right evaluation of pairs. Equally well, one could define

$$A \mathbin{\overleftarrow{\times}} B = \mathsf{E}(\lambda b.\, \mathsf{E}(\lambda a.\, \mathsf{r}(a, b))A)B,$$

which corresponds to right-to-left evaluation.

In the powerset example, we obtain $A \mathbin{\vec{\times}} B = \bigcup_{a \in A} \bigcup_{b \in B} \{(a, b)\}$, which is just the ordinary product set $A \times B$. The dual product $A \mathbin{\overleftarrow{\times}} B$ leads to the same result. We call strong monads where $\vec{\times} = \overleftarrow{\times}$ *symmetric*.

We now present the properties of the two standard products.

Theorem 3.1

With every strong monad in a CCC, for both $\bar{\times} = \overrightarrow{\times}$ and $\bar{\times} = \overleftarrow{\times}$, the following holds:

(P1) For all a in X and B in MY, $\mathsf{r}a \mathbin{\bar{\times}} B = \mathsf{M}\sigma_a B$, where $\sigma_a b = (a, b)$.
For all A in MX and b in Y, $A \mathbin{\bar{\times}} \mathsf{r}b = \mathsf{M}\sigma^b A$, where $\sigma^b a = (a, b)$.

(P2) (Naturality)
For all A in MX, B in MY, $f : X \to X'$, and $g : Y \to Y'$:
$\mathsf{M}fA \mathbin{\bar{\times}} \mathsf{M}gB = \mathsf{M}(f \times g)(A \mathbin{\bar{\times}} B)$, where $(f \times g)(a, b) = (fa, gb)$.

(P3) (Associativity) For A in MX, B in MY, and C in MZ:
$A \mathbin{\bar{\times}} (B \mathbin{\bar{\times}} C) = \mathsf{M}\alpha\,((A \mathbin{\bar{\times}} B) \mathbin{\bar{\times}} C)$, where $\alpha((a, b), c) = (a, (b, c))$.

If the strong monad is symmetric, then $\bar{\times} = \overrightarrow{\times} = \overleftarrow{\times}$ additionally satisfies

(P4) (Commutativity) For A in MX and B in MY:
$B \mathbin{\bar{\times}} A = \mathsf{M}\gamma\,(A \mathbin{\bar{\times}} B)$, where $\gamma(a, b) = (b, a)$.

Proof: Statements (P1) through (P3) can be shown by equational reasoning in the λ-calculus using the axioms of extension.

For (P4), note that $B \mathbin{\overrightarrow{\times}} A = \mathsf{M}\gamma\,(A \mathbin{\overleftarrow{\times}} B)$ holds in general. Thus, '$\overrightarrow{\times}$' is commutative iff '$\overleftarrow{\times}$' is commutative iff the strong monad is symmetric. □

If we want to augment a given strong monad (M, r, E) with a product operation '$\bar{\times}$' of our choice, then '$\bar{\times}$' should at least satisfy (P1) through (P3) to be as good as the standard products.

4 Power Domain Constructions from Non-Deterministic Theories

We now assume that the functional language with computations, whose semantics shall be described by a strong monad, is a non-deterministic language with a binary choice operator $\mid$, a special expression `fail`, and syntactic means for recursion.

The choice operator is open for various different interpretations. We may think that encountering an expression $A \mid B$, the interpreter of the language employs some hidden strategy to choose one of A and B for evaluation, or chooses one of them at random with equal probability, or evaluates both A and B in turn and displays all resulting values (similar to a Prolog interpreter). In any circumstances, we need a semantic function $\oplus : MX \times MX \to MX$ to define $[\![A \mid B]\!] = [\![A]\!] \oplus [\![B]\!]$. (In a typed language, the two branches of a choice should have the same type.)

An evaluation of the expression `fail` immediately terminates without producing a result. Assuming that this expression can have any type, we need a semantic constant $\diamond : MX$ for every X to define $[\![\mathtt{fail}]\!] = \diamond$.

In order to model recursion, the semantic domains should finally admit some kind of fixed point formation. To do so, we assume that the semantic monad operates within the category

$\mathcal{DCPO}$, and that all semantic domains MX actually be cpo's with least element $\perp$. This $\perp$ will then be the semantics of computations that do not yield ordinary results because of non-termination. We do not assume that X itself contains $\perp$; e.g., the domain of integer values is $\{\ldots, -1, 0, 1, \ldots\}$ with the discrete order. We call semantic domains with $\perp$, $\diamond$, and '$\oplus$' *power domains*, and strong monads M which produce power domains MX *power domain constructions.*

Our way to obtain power domain constructions is to set up an algebraic theory with operators $\perp$, $\diamond$, and '$\oplus$' and appropriate axioms of our choice, and then take the free construction in $\mathcal{DCPO}$ w.r.t. this theory. This approach has a long-standing tradition. It was first suggested in [5], then extended in [11], studied more closely in [6] and later [2, 3], and was largely used in [4].

Usually, an algebraic theory consists of a set of operators with given arity and a set of axioms in the form of equations $L = R$ over these operators. Since we want to consider algebraic theories in the category $\mathcal{DCPO}$, we also allow inequations $L \leq R$ as axioms. Also, we allow arities of the form $D \times X^n \to X$, where D is a fixed dcpo, and X is the algebra. In $\mathcal{SET}$, this would give no further generalization, since an operation $f : D \times X^n \to X$ may be replaced by a family $f_d : X^n \to X$ with d in D. In $\mathcal{DCPO}$, this reduction is not possible, because $f : D \times X^n \to X$ stipulates continuity of f in its first argument.

A *non-deterministic theory* or briefly *ND-theory* has two nullary operators $\perp$ and $\diamond$, a binary operator '$\oplus$', and an arbitrary set of axioms, which actually are equations or inequations, or are equivalent to those after introducing some auxiliary operations (an example for the latter case is the probabilistic theory introduced below). An ND-theory should contain the property $\perp \leq x$, as an explicit axiom or as a theorem, to guarantee that $\perp$ will be the least element. The actual choice of the axioms depends on the interpretation of the non-determinism and the number of distinctions we wish to make about the result of computations.

In the remainder of this section, we present some sample ND-theories. First assume that we are interested in the *set* of values a non-deterministic expression may produce. Then we should give '$\oplus$' the properties of set union: associativity $(A \oplus B) \oplus C = A \oplus (B \oplus C)$, commutativity $A \oplus B = B \oplus A$, and idempotence $A \oplus A = A$. A failing expression produces no values at all; thus, $\diamond$ is the neutral element of '$\oplus$': $A \oplus \diamond = A$. A non-terminating expression produces no values, too, whence $\perp = \diamond$. We call this the *lower set-like theory* LS. It is well-known that the free construction for LS is the *lower* or *Hoare power domain* [5].

If we want to retain the information whether a computation is possibly non-terminating, we have to distinguish between $\perp$ and $\diamond$. This is done simply by dropping the axiom $\perp = \diamond$. The element $\diamond$ is still the neutral element of '$\oplus$', since the expressions `fail | ` A and A may possibly produce the same values and have the same termination behaviour. We call this theory the *set-like theory* S. Its free construction is none of the classical power constructions. The internal structure of its power domains is studied in [4].

If we want to retain the information, how often a particular value may occur as the result of an expression, then we are interested in the bag or multiset of possible result values, and we have to drop the axiom of idempotence $A \oplus A = A$. According to whether we keep $\perp = \diamond$ or not, the resulting theory is the *lower multiset-like theory* LM or the *multiset-like theory* M. The power domains of LM and M have not been investigated much so far. Although general idempotence does not hold in LM and M, we still have $\diamond \oplus \diamond = \diamond$ by neutrality, and also $\perp \oplus \perp = \perp$, since $\perp \leq \perp \oplus \perp \leq \diamond \oplus \perp = \perp$.

Quite a different theory results from a probabilistic interpretation of non-determinism, where $A \oplus B$ means that A and B are chosen with equal probability 1/2. Then '$\oplus$' is commutative and idempotent, but not associative, since in $A \oplus (B \oplus C)$, A is chosen with probability 1/2, whereas in $(A \oplus B) \oplus C$, it has probability 1/4. On the other hand, there is still something left of associativity: the *rectangle axiom* $(A \oplus B) \oplus (C \oplus D) = (A \oplus C) \oplus (B \oplus D)$, which holds since all four components are chosen with probability 1/4. The four operands A, B, C, and D may be thought as arranged at the corners of a rectangle; the axiom then says that they may be combined first horizontally, then vertically, as well as first vertically, then horizontally; hence the name. The rectangle law also holds in the theories S and M as a consequence of commutativity and associativity.

There is another, less obvious axiom for the probabilistic interpretation. In a recursive definition such as `x = 1 | x`, the result 1 is chosen with probability 1/2 in the first iteration, with probability 1/4 in the second, and so on. Altogether, it is chosen with probability 1, and the probability of non-termination is 0. Generalizing, this leads to the axiom that the fixed point iteration for $X = A \oplus X$ yields A, for every A. This fixed-point axiom $\mu X.\, A \oplus X = A$ is equivalent to $A \oplus A = A$, which is idempotence, together with the conditional axiom $A \oplus B = B \Rightarrow A \leq B$. The fixed-point axiom can be reduced to a set of equations by introducing an auxiliary operation $f : \mathbb{N}_0^\infty \to X$, where $\mathbb{N}_0^\infty$ is the dcpo $\{1 < 2 < \cdots < \infty\}$. With this operation, the fixed point iteration in $\mu X.\, A \oplus X = A$ can be modelled by the equations $f(0) = \perp$, $f(n+1) = A \oplus f(n)$, and $f(\infty) = A$.

Under the probabilistic interpretation, $\diamond$ is by no means a neutral element. Thus, we obtain as *probabilistic theory* P the theory with commutativity, idempotence, rectangle axiom, and fixed point axiom. The theory with additionally $\perp = \diamond$ is called the *lower probabilistic theory* LP. We conjecture that LP is equivalent to the theory of probabilistic algebras in [7, 8] at least in the case of continuous dcpo's, so that for these dcpo's, the free construction for LP would coincide with the one studied there.

The basic properties that are shared by all theories presented above are the rectangle law, $\diamond \oplus \diamond = \diamond$, and $\perp \oplus \perp = \perp$. ND-theories with these properties will be called *decent*.

5 Free Constructions

Let T be an ND-theory. A *model* of T in the category $\mathcal{DCPO}$, or simply a *T-model*, is a power domain (a dcpo with $\perp$, $\diamond$, and '$\oplus$'), whose operations satisfy the axioms of T.

A T-algebra homomorphism, or shortly *T-morphism*, is a *continuous* function between T-models preserving all operations. T-morphisms are thus continuous, strict ($f\perp = \perp$), $\diamond$-strict ($f\diamond = \diamond$), and additive ($f(A \oplus B) = fA \oplus fB$).

Definition 5.1 (Free constructions) A *free construction* $(\mathcal{P}, \mathsf{r})$ for an algebraic theory T maps every dcpo X into a T-model $\mathcal{P}X$ with a morphism $\mathsf{r} : X \to \mathcal{P}X$, such that for every T-model A and every morphism $f : X \to A$, there is a unique T-morphism $\bar{f} : \mathcal{P}X \to A$ with $\bar{f} \circ \mathsf{r} = f$.

Categorically speaking, free constructions for T are left adjoint to the forgetful functor from the category of T-models and T-morphisms to $\mathcal{DCPO}$. Free constructions are uniquely determined up to isomorphism. They also exist:

Theorem 5.2 In $\mathcal{DCPO}$, free constructions exist for all algebraic theories.

Proof: The proof idea appears in [6], where the theorem is proved for one particular algebraic theory. This idea is then applied in [2, 3] to a larger class of theories. The same basic idea also applies in generality. □

Free constructions for ND-theories T are power domain constructions: Since $\mathcal{P}Y$ is a T-model, every function $f : X \to \mathcal{P}Y$ has a (unique) extension $\mathsf{E}f = \bar{f}$, which is strict, $\diamond$-strict, and additive, and satisfies (E1) $\mathsf{E}f \circ \mathsf{r} = f$. The only properties which remain to be shown are (E2), (E3), and continuity of E. To do so, we collect some facts about free algebras.

Lemma 5.3 Let $(\mathcal{P}, \mathsf{r})$ be a free construction for theory T, and let X be a dcpo. Let B be a subset of $\mathcal{P}X$ that is closed w.r.t. all operations of T and directed joins, and that contains all generators $\mathsf{r}x$. Then B equals $\mathcal{P}X$.

Proof: See [4], Lemma 4.4. □

From this Lemma, we conclude:

Proposition 5.4 Let $\mathcal{P}$ be the free power construction for an ND-theory T, and X and Y be dcpo's with a binary monotonic operation $\oplus : Y \times Y \to Y$. Let F and G be two morphisms from $\mathcal{P}X$ to Y, which are both additive, and can be compared on generators ($F \circ \mathsf{r} \le G \circ \mathsf{r}$) as well as on empty elements ($F\diamond \le G\diamond$ and $F\bot \le G\bot$). Then $F \le G$.

The same holds if '$\le$' is replaced by '$=$' throughout.

Proof: Apply Lemma 5.3 to $B = \{A \in \mathcal{P}X \mid FA \le GA\}$. □

Note that in Prop. 5.4, Y need not be a T-model; in particular, it is not required to have $\bot$ and $\diamond$, and its operation '$\oplus$' need not satisfy any axioms; not even continuity of '$\oplus$' is required.

The proposition may be used to prove (E2), (E3), and continuity of E, as shown explicitly in [2]. Later, we need a generalization of the proposition:

Theorem 5.5 Let $\mathcal{P}$ be the free power construction for an ND-theory T, let $X_1, \ldots, X_n$ and Y be dcpo's with a binary monotonic operation $\oplus : Y \times Y \to Y$. Let F and G be two morphisms from $\mathcal{P}X_1 \times \cdots \times \mathcal{P}X_n$ to Y with the following properties:

(1) F and G are both additive in every argument, if the remaining arguments are kept fixed.

(2) If at least one of the A_i is of form $\mathsf{r}a_i$,
then $F(A_1, \ldots, A_n) \le G(A_1, \ldots, A_n)$.

(3) If all A_i are empty (i.e., $\diamond$ or $\bot$), then $F(A_1, \ldots, A_n) \le G(A_1, \ldots, A_n)$.

Then $F \le G$ can be concluded.

The same holds if '$\le$' is replaced by '$=$' throughout.

Proof: This is proved by induction on n. For $n = 1$, this is Proposition 5.4. For $n > 1$, we use Proposition 5.4 to show $F' \le G'$, where $F', G' : \mathcal{P}X_1 \to [\mathcal{P}X_2 \times \cdots \times \mathcal{P}X_n \to Y]$ are curried versions of F and G. By (1), both F' and G' are additive. By (2), $F' \circ \mathsf{r} \le G' \circ \mathsf{r}$ holds. The two functions $F'\diamond$ and $G'\diamond$ have $n-1$ arguments, and $F'\diamond \le G'\diamond$ holds by induction hypothesis. The same is true for $F'\bot$ and $G'\bot$. □

6 The Standard Products in Free Power Constructions

In this section, we look at the special properties of the standard products in free power constructions. Particularly strong statements are possible in the case of free constructions w.r.t. decent theories. Remember that an ND-theory is decent if it includes the rectangle axiom and the two properties $\diamond \oplus \diamond = \diamond$ and $\bot \oplus \bot = \bot$. The significance of this notion is indicated by the following proposition:

Proposition 6.1 Let T be a decent ND-theory. If X is a power domain and Y a T-model, then the (pointwise defined) sum of additive (strict, $\diamond$-strict) functions from X to Y is additive (strict, $\diamond$-strict) again. The constant functions $\lambda x.\,\bot$ and $\lambda x.\,\diamond$ are additive.

Proof: Let f and g be additive, and let $h = f \oplus g = \lambda x.\, fx \oplus gx$. Then

$$\begin{aligned} h(a \oplus b) &= f(a \oplus b) \oplus g(a \oplus b) &&= (fa \oplus fb) \oplus (ga \oplus gb) \\ &= (fa \oplus ga) \oplus (fb \oplus gb) &&= ha \oplus hb \end{aligned}$$

whence h is additive. The remaining statements are simple consequences of $\bot \oplus \bot = \bot$ and $\diamond \oplus \diamond = \diamond$. □

We now come to the properties of the standard products in free constructions.

Proposition 6.2 Let T be an ND-theory and $\mathcal{P}$ its free construction.

(1) The left product '$\overrightarrow{\times}$' is strict, $\diamond$-strict, and additive in its left argument:
$\bot \overrightarrow{\times} B = \bot \qquad \diamond \overrightarrow{\times} B = \diamond \qquad (A_1 \oplus A_2) \overrightarrow{\times} B = (A_1 \overrightarrow{\times} B) \oplus (A_2 \overrightarrow{\times} B)$.

(2) The right product '$\overleftarrow{\times}$' is strict, $\diamond$-strict, and additive in its right argument.

(3) If T is decent, then both '$\overrightarrow{\times}$' and '$\overleftarrow{\times}$' are additive in both arguments.

Proof: We concentrate on '$\overrightarrow{\times}$'. Remember $A \overrightarrow{\times} B = \mathsf{E}(\lambda a.\, \mathsf{E}(\lambda b.\, \mathsf{r}(a,b))B)A$. Thus (1) follows from the fact that extended functions $\mathsf{E}f$ are always strict, $\diamond$-strict, and additive.

For (3), we compare $FA = A \overrightarrow{\times} (B_1 \oplus B_2)$ and $GA = (A \overrightarrow{\times} B_1) \oplus (A \overrightarrow{\times} B_2)$ by Prop. 5.4. By (1), F is additive, strict, and $\diamond$-strict. By Prop. 6.1, G is additive, strict, and $\diamond$-strict as sum of two such functions. By (P1), $F(\mathsf{r}a) = \mathsf{M}\sigma_a(B_1 \oplus B_2)$ and $G(\mathsf{r}a) = \mathsf{M}\sigma_a B_1 \oplus \mathsf{M}\sigma_a B_2$ hold. The function $\mathsf{M}\sigma_a = \mathsf{E}(\mathsf{r} \circ \sigma_a)$ is additive. Thus, $F(\mathsf{r}a)$ equals $G(\mathsf{r}a)$. □

Next, we consider symmetry.

Proposition 6.3 Let T be an ND-theory with free construction $\mathcal{P}$.

(1) If $\mathcal{P}$ is symmetric, then $\bot = \diamond$ holds in all $\mathcal{P}$-powerdomains.

(2) If T is decent with $\bot = \diamond$, then $\mathcal{P}$ is symmetric.

Proof: By Prop. 6.2 (1) and (2), $\bot \mathbin{\vec{\times}} \diamond = \bot$ and $\bot \mathbin{\overleftarrow{\times}} \diamond = \diamond$ hold. If $\mathcal{P}$ is symmetric, then '$\vec{\times}$' and '$\overleftarrow{\times}$' coincide, whence $\bot = \diamond$.

For (2), we apply Theorem 5.5 to compare $A \mathbin{\vec{\times}} B$ and $A \mathbin{\overleftarrow{\times}} B$ as functions of two arguments. By Prop. 6.2 (3), both functions are additive in both arguments. By (P1), they yield equal results if $A = \mathsf{r}a$ or $B = \mathsf{r}b$. By $\bot = \diamond$, they coincide if both A and B are empty. □

Thus, in particular the power constructions $\mathcal{P}_{LS}$, $\mathcal{P}_{LM}$, and $\mathcal{P}_{LP}$ are symmetric, whereas $\mathcal{P}_S$, $\mathcal{P}_M$, and $\mathcal{P}_P$ are not symmetric. To speak precisely, the non-symmetry statement is only proved after it is shown that the axioms of the respective theories do not imply $\bot = \diamond$. This can be done by presenting models with $\bot \neq \diamond$. The dcpo $\{\bot < \diamond\}$ is an M- and S-model with '$\oplus$' being join, and a P-model with '$\oplus$' being meet.

7 The Choice Product

So far, we have only seen standard products. Now, we try the most obvious non-standard product in a power domain construction: the product $A * B = A \mathbin{\vec{\times}} B \oplus A \mathbin{\overleftarrow{\times}} B$, where we choose the evaluation order non-deterministically. To avoid bothering about two different versions of '$*$', we assume that '$\oplus$' is commutative. We shall soon see that idempotence is also important, and finally we assume the rectangle law to make the results of the previous section available. Thus, the following results cover the theories S and P, but not M.

Theorem 7.1 Let T be an ND-theory, where '$\oplus$' is commutative, idempotent, and the rectangular law holds. Then '$*$' has the following properties:

(P1) $\mathsf{r}a * B = \mathsf{M}\sigma_a B$ and $A * \mathsf{r}b = \mathsf{M}\sigma^b A$.

(P2) Naturality: $\mathsf{M}fA * \mathsf{M}gB = \mathsf{M}(f \times g)(A * B)$.

(P4) Commutativity: $B * A = \mathsf{M}\gamma\,(A * B)$.

- Biadditivity: '$*$' is additive in both arguments.
- $\diamond * \diamond = \diamond$ and $\bot * \bot = \bot$.

Proof: By Prop. 6.1, '$*$' is biadditive as the sum of '$\overleftarrow{\times}$' and '$\vec{\times}$', which are biadditive by Prop. 6.2. The other statements can be shown by simple equational reasoning. □

The reader will have noticed that associativity (P3) does not occur in the theorem above. The reason is that it surprisingly does not hold in general. Using biadditivity, $A * (B * C)$ can be expanded into

$$(A \mathbin{\vec{\times}} (B \mathbin{\vec{\times}} C) \oplus A \mathbin{\vec{\times}} (B \mathbin{\overleftarrow{\times}} C)) \oplus (A \mathbin{\overleftarrow{\times}} (B \mathbin{\vec{\times}} C) \oplus A \mathbin{\overleftarrow{\times}} (B \mathbin{\overleftarrow{\times}} C)),$$

whereas $(A * B) * C$ becomes

$$((A \mathbin{\vec{\times}} B) \mathbin{\vec{\times}} C \oplus (A \mathbin{\overleftarrow{\times}} B) \mathbin{\vec{\times}} C) \oplus ((A \mathbin{\vec{\times}} B) \mathbin{\overleftarrow{\times}} C \oplus (A \mathbin{\overleftarrow{\times}} B) \mathbin{\overleftarrow{\times}} C).$$

It seems as if the two sums could be related after switching the two middle summands by the rectangle law. But this impression is wrong; the first sum contains evaluation orders ABC, ACB, BCA, and CBA, whereas the second sum contains the orders ABC, BAC, CAB, and CBA. Thus, both sums contain a different selection of 4 evaluation orders from the altogether 6 possible evaluation orders for a triple.

Of course, this does not prove that the associativity of '$*$' really fails. Indeed, it *does* hold in the set-like theory S, which will be seen in Section 8, but fails in the probabilistic theory P.

To show non-associativity of '$*$' in P, we need a concrete P-model which can be used to distinguish members of the free P-models $\mathcal{P}X$, which are merely abstractly described by their universal property. Let $\mathbb{I}$ be the dcpo of real numbers from 0 to 1, which becomes a P-model by defining $\perp = 0$, $\diamond = 1$, and $a \oplus b = (a+b)/2$. The operation '$\oplus$' is well-defined and continuous, and obviously commutative, idempotent, and satisfies the rectangular law. The fixed point axiom holds, since equations $x = a \oplus x$ have exactly one solution, namely $x = a$.

Since $\mathbb{I}$ is a P-model, the map $\lambda x.\, \perp : X \to \mathbb{I}$ has a (unique) extension $\varphi : \mathcal{P}X \to \mathbb{I}$ with $\varphi\perp = 0$ and $\varphi\diamond = 1$. Intuitively, φA is the probability that the computation A fails without result.

We compare $L = \perp * (\perp * \diamond)$ and $R = \mathsf{M}\alpha\,((\perp * \perp) * \diamond)$. First, $\perp * \diamond = \perp \overrightarrow{\times} \diamond \oplus \perp \overleftarrow{\times} \diamond = \perp \oplus \diamond$ holds. Thus, using additivity, we obtain

$$\begin{array}{lcl} L &=& \perp * (\perp \oplus \diamond) \;=\; (\perp * \perp) \oplus (\perp * \diamond) \;=\; \perp \oplus (\perp \oplus \diamond) \qquad \text{and} \\ R &=& \mathsf{M}\alpha\,(\perp * \diamond) \;=\; \mathsf{M}\alpha\,(\perp \oplus \diamond) = \perp \oplus \diamond. \end{array}$$

Therefore, $\varphi L = 0 \oplus (0 \oplus 1) = 1/4$, whereas $\varphi R = 0 \oplus 1 = 1/2$, whence $L \neq R$ follows.

8 Decent Products

After the attempt to construct an alternative product by an explicit formula has failed to yield an associative product, we try an axiomatic approach. We call a binary operation $\bar{\times} : \mathcal{P}X \times \mathcal{P}Y \to \mathcal{P}(X \times Y)$ a *decent product* iff it is biadditive, satisfies (P1), $\diamond \,\bar{\times}\, \diamond = \diamond$, and $\perp \bar{\times} \perp = \perp$. From Theorem 5.5, we can conclude immediately:

Proposition 8.1 Let '$\bar{\times}_1$' and '$\bar{\times}_2$' be two decent products. Then $\bar{\times}_1 \leq \bar{\times}_2$ iff $\diamond \,\bar{\times}_1 \perp \leq \diamond \,\bar{\times}_2 \perp$ and $\perp \bar{\times}_1 \diamond \leq \perp \bar{\times}_2 \diamond$. The same holds with '$\leq$' replaced by '$=$'.

Thus, a decent product is uniquely determined by its values on $(\perp, \diamond)$ and $(\diamond, \perp)$. We are particularly interested in the following four possibilities:

- Left product $\times_L$: $\quad \perp \times_L \diamond = \perp \qquad \diamond \times_L \perp = \diamond$
- Right product $\times_R$: $\quad \perp \times_R \diamond = \diamond \qquad \diamond \times_R \perp = \perp$
- Strict product $\check{\times}$: $\quad \perp \check{\times} \diamond = \perp \qquad \diamond \,\check{\times} \perp = \perp$
- Parallel product $\hat{\times}$: $\quad \perp \hat{\times} \diamond = \diamond \qquad \diamond \,\hat{\times} \perp = \diamond$

We have already met some of these products:

Proposition 8.2 For a decent theory, the left and right products exist and are given by the standard products '$\overrightarrow{\times}$' and '$\overleftarrow{\times}$'. For the set-like theory S, the strict product exists and is given by the choice product.

Proof: The standard products satisfy (P1) by Theorem 3.1, and are biadditive and yield the required results for empty arguments by Prop. 6.2. The choice product is decent by Prop. 7.1. In case of theory S, the required behaviour on empty arguments can be easily checked. $\square$

The properties of left and right product are thus given by the properties of the standard products. The properties of strict and parallel product are as follows:

Theorem 8.3 For every ND-theory T, the strict product $\check{\times}$ and the parallel product $\hat{\times}$ — provided they exist — are biadditive, satisfy (P1), are natural (P2), associative (P3), and commutative (P4).

Proof: Biadditivity is part of the definition (strict and parallel product are decent). To prove (P2), (P3), and (P4), use Theorem 5.5. □

The strict product is really strict, and the parallel product is $\diamond$-strict:

Proposition 8.4

(1) If $\bot \oplus \bot = \bot$ holds in T and '$\check{\times}$' exists, then $A \mathbin{\check{\times}} \bot = \bot \mathbin{\check{\times}} B = \bot$.

(2) If $\diamond \oplus \diamond = \diamond$ holds in T and '$\hat{\times}$' exists, then $A \mathbin{\hat{\times}} \diamond = \diamond \mathbin{\hat{\times}} B = \diamond$.

Proof: Compare $F = \lambda A.\, A \mathbin{\check{\times}} \bot$ and $G = \lambda A.\, \bot$ by Prop. 5.4. Proceed analogously for '$\hat{\times}$'. □

9 Existence of Parallel Products

In this section, we show the existence of parallel products for the ND-theories S, M, and P. This is done by a universal construction, which in principle works for all decent ND-theories. The idea is to construct '$\hat{\times}$' by uncurrying a function from $\mathcal{P}X$ to (a sub-dcpo of) $[\mathcal{P}Y \to \mathcal{P}(X \times Y)]$. This function is obtained by extending a suitable function from X. An analogous construction may be used to show the existence of strict products for these theories.

In the sequel, let T be a decent ND-theory, and $\mathcal{P}$ the corresponding free construction. For dcpo's X and Y, let $Z = [\mathcal{P}Y \overset{\oplus\diamond}{\to} \mathcal{P}(X \times Y)]$ be the dcpo of *additive $\diamond$-strict* morphisms from $\mathcal{P}Y$ to $\mathcal{P}(X \times Y)$. We make Z into a T-model by defining '$\oplus$' and $\diamond$ pointwise as $f \oplus g = \lambda B.\, fB \oplus gB$ and $\diamond = \lambda B.\, \diamond$, and additionally $\bot = \lambda B.\, \bot \mathbin{\bar{\times}} B$.

The operations are well defined: By Prop. 6.1, $f \oplus g$ is additive and $\diamond$-strict, and the function $\diamond$ is additive. It is obviously $\diamond$-strict. Finally, the function $\bot$ is additive and $\diamond$-strict, since '$\bar{\times}$' is additive and $\diamond$-strict in its right argument.

Next, we have to show that Z is a T-model. We cannot do this in general, because the theory T may contain exotic axioms that do not carry over to the function space. On the other hand, all axioms that occur in this paper do carry over, as is shown in the following discussion.

Equational and inequational axioms with '$\oplus$' as the only operation hold in Z because '$\oplus$' on functions is defined pointwise. This covers commutativity, associativity, idempotence, and the rectangle axiom, and even the fixed point axiom of the theory P, which is equivalent to $A \oplus A = A$ (idempotence) and $A \oplus B = B \Rightarrow A \leq B$.

Similarly, axioms involving $\diamond$ such as $A \oplus \diamond = A$ and $\diamond \oplus \diamond = \diamond$ hold in Z because of the pointwise definition of the function $\diamond$.

The problem is with axioms involving $\perp$, since this function is not defined pointwise. For $\perp \leq A$ for instance, we have to compare $\lambda B. \perp \overleftarrow{\times} B$ with an arbitrary additive $\diamond$-strict function f by Prop. 5.4. Both functions are additive and $\diamond$-strict. For $\perp$, $\perp \overleftarrow{\times} \perp = \perp \leq f\perp$ holds. For generators, we have $\perp \overleftarrow{\times}\, \mathsf{r}b = \mathsf{M}\sigma^b \perp = \perp \leq f(\mathsf{r}b)$.

For the axiom $\perp = \diamond$, we know from Prop. 6.3 that decent theories with this axiom induce symmetric power constructions, whence $\perp \overleftarrow{\times} B = \perp \overrightarrow{\times} B = \perp = \diamond$.

Once Z is shown to be a T-model, we introduce the function $p = \lambda a.\, \mathsf{M}\sigma_a$, which maps from X to Z, since $\mathsf{M}\sigma_a = \mathsf{E}(\mathsf{r} \circ \sigma_a)$ is additive and $\diamond$-strict as an extended function. Extending p yields $P : \mathcal{P}X \to Z$, which by uncurrying induces a function $\hat{\times} : \mathcal{P}X \times \mathcal{P}Y \to \mathcal{P}(X \times Y)$. We claim that '$\hat{\times}$' is the parallel product for $\mathcal{P}$.

The function '$\hat{\times}$' is additive and $\diamond$-strict in its left argument, since P is the extension of p, extended functions are additive and $\diamond$-strict, and '$\oplus$' and $\diamond$ are defined pointwise in Z. It is additive and $\diamond$-strict in its right argument, since $\lambda B.\, A \hat{\times} B = PA$ is in Z. Considering $\perp$, $\perp \hat{\times} \perp = \perp \overleftarrow{\times} \perp = \perp$ holds.

Finally, we show (P1). One part is easy: $\mathsf{r}a \hat{\times} B = P(\mathsf{r}a)B = paB = \mathsf{M}\sigma_a B$ holds. For the other part, we compare $\lambda A.\, A \hat{\times} \mathsf{r}b$ and $\mathsf{M}\sigma^b$ by Prop. 5.4. Both functions are additive and $\diamond$-strict. For generators, $\mathsf{r}a \hat{\times} \mathsf{r}b = \mathsf{M}\sigma_a(\mathsf{r}b) = \mathsf{r}(a, b) = \mathsf{M}\sigma^b(\mathsf{r}a)$ holds, and for $\perp$, we have $\perp \hat{\times} \mathsf{r}b = \perp \overleftarrow{\times} \mathsf{r}b = \mathsf{M}\sigma^b \perp$ by (P1) of '$\overleftarrow{\times}$'.

Summarizing, we have shown the existence of parallel products for the theories S, M, and P. If T is another decent theory, then our method also works if the axioms of T can be proved for the function space Z.

10 Conclusion

To define the semantics of pairing in Moggi's framework, a product operation $\bar{\times} : MX \times MY \to M(X \times Y)$ for the computational strong monad M is needed. After having studied the algebraic properties of the two standard products '$\overrightarrow{\times}$' and '$\overleftarrow{\times}$', which correspond to left-to-right and right-to-left evaluation, we have looked for possible alternative products. In the special case, where M is given by the free construction for a decent non-deterministic theory, we found two such alternative products: the strict product '$\check{\times}$' and the parallel product '$\hat{\times}$'.

Acknowledgments

This paper was written during my one-year visit at Imperial College, London. The visit was made possible by a grant of the Deutsche Forschungsgemeinschaft. I have to thank all the people of the TFM section of Imperial College for their stimulating environment, and for giving me the opportunity to participate in their Section Workshop.

References

[1] C.A. Gunter. Relating total and partial correctness interpretations of non-deterministic programs. In P. Hudak, editor, *Principles of Programming Languages (POPL '90)*, pages 306–319. ACM, 1990.

[2] R. Heckmann. *Power Domain Constructions.* PhD thesis, Universität des Saarlandes, 1990.

[3] R. Heckmann. Power domain constructions. *Science of Computer Programming*, 17:77–117, 1991.

[4] R. Heckmann. Power domains supporting recursion and failure. In J.-C. Raoult, editor, *CAAP '92*, pages 165–181. *Lecture Notes in Comp. Sc. 581*, Springer-Verlag, 1992.

[5] M.C.B Hennessy and G.D. Plotkin. Full abstraction for a simple parallel programming language. In J. Becvar, editor, *Foundations of Computer Science*, pages 108–120. *Lecture Notes in Comp. Sc. 74*, Springer-Verlag, 1979.

[6] R. Hoofman. Powerdomains. Technical Report RUU-CS-87-23, Rijksuniversiteit Utrecht, November 1987.

[7] C.J. Jones. *Probabilistic Non-Determinism.* PhD thesis, University of Edinburgh, 1990.

[8] C.J. Jones and G.D. Plotkin. A probabilistic powerdomain of evaluations. In *LICS '89*, pages 186–195. IEEE Computer Society Press, 1989.

[9] J. Lambek and P. J. Scott. *Introduction to Higher Order Categorical Logic*, volume 7 of *Cambridge Studies in Advanced Mathematics.* Cambridge University Press, 1986.

[10] S. Mac Lane. *Categories for the Working Mathematician.* Springer-Verlag, 1971.

[11] M.G. Main. Free constructions of powerdomains. In A. Melton, editor, *Mathematical Foundations of Programming Semantics*, pages 162–183. *Lecture Notes in Comp. Sc. 239*, Springer-Verlag, 1985.

[12] E. Moggi. Computational lambda-calculus and monads. In *4th LICS Conference*, pages 14–23. IEEE, 1989.

[13] E. Moggi. Notions of computation and monads. *Information and Computation*, 93:55–92, 1991.

[14] G.D. Plotkin. A powerdomain construction. *SIAM Journal on Computing*, 5(3):452–487, 1976.

[15] M.B. Smyth. Power domains. *Journal of Computer and System Sciences*, 16:23–36, 1978.