Reasoning about Probability and Nondeterminism

Faris Abou-Saleh Kwok-Ho Cheung Jeremy Gibbons
Department of Computer Science, University of Oxford
firstname.lastname@cs.ox.ac.uk

1. Introduction

Probabilistic and nondeterministic choice are two standard examples of computational effect, and it is important for some problems to be able to use them in combination—for example, to model probabilistic systems that depend on nondeterministic inputs. However, the algebraic properties that characterise their interaction are tricky to get right (and we have ourselves got them wrong in the past [2]). We outline the problem, and present a technique for diagrammatic reasoning about their properties.

2. Algebraic effects

In order to express computational effects in a pure functional language, we need some way of separating the *syntactic* specification of an effectful computation from the *semantic* considerations involved in executing that computation. One way of doing that is via *algebraic effects* and *effect handlers* [6, 7].

An algebraic effect is a syntactic constructor for terms representing computations. For example, if we wanted to introduce the computational effect of nondeterminism into a pure program, we could do so by defining an algebraic effect $\Box$, a binary constructor on terms, so that $(1 \Box 2) \Box 3$ represents a nested nondeterministic choice between three outcomes $1,2,3$. One can think of the resulting terms as binary trees, with pure values at the leaves and choices at the branches. One might assert some axioms, stating that certain distinctions between terms are irrelevant. For example, we might assert that $\Box$ is associative; then we would treat $1 \Box (2 \Box 3)$ as an equivalent term—that is, we quotient the set of terms by the equivalence induced by the axioms, and we take equality only up to this equivalence (i.e., the terms are effectively lists). We might additionally assert commutativity and idempotence; then the terms denote finite sets. An *algebraic theory* consists of a collection of operators (such as $\Box$) and axioms (such as associativity, commutativity, and idempotence); a *model* of the theory is a carrier set with implementations of the operators that satisfy the axioms; an *effect handler* is a function that interprets a term in a model.

Composition of effectful computations, written $\gg\!=$, is a matter of substituting subsequent subterms for the leaves of the initial term. For example, if computation $m = 4 \Box 7$ has two possible outcomes, and continuation $k\ x = x \Box (x+1)$ yields two more outcomes for a given input, then their composition $m \gg\!= k$ is the term $(4 \Box 5) \Box (7 \Box 8)$ with four outcomes, obtained by substituting $k\,4$ for 4 and $k\,7$ for 7 in m. Nondeterministic choice is called an *algebraic* effect because composition distributes over it from the right: $(m \Box n) \gg\!= k = (m \gg\!= k) \Box (n \gg\!= k)$.

Finitely supported probability distributions are a model of another algebraic theory, with an indexed family of binary operators $\lhd\rhd$; the idea is that $m \lhd w \rhd n$ represents the computation that behaves like m with probability $w \in 0\,..\,1$ and like n with probability $\bar{w} = 1 - w$. So a weighted coin toss is $coin\ w = True \lhd w \rhd False$. For axioms, we have two identity laws $m \lhd 0 \rhd n = n$ and $m \lhd 1 \rhd n = m$, idempotence $m \lhd w \rhd m = m$, quasi-commutativity $m \lhd w \rhd n = n \lhd \bar{w} \rhd m$, quasi-associativity $m \lhd w \rhd (n \lhd x \rhd p) = (m \lhd y \rhd n) \lhd z \rhd p$ where $w = yz \wedge \bar{z} = \bar{w}\bar{x}$, and algebraicity again: $m \lhd w \rhd n \gg\!= k = (m \gg\!= k) \lhd w \rhd (n \gg\!= k)$.

3. Combining effects

Algebraic theories combine very naturally: one simply unions the signatures and axioms, perhaps adding some additional axioms to describe how the effects interact. Finding a model of the composite theory is not so simple, however; it is not in general a combination of models of the parts. The combination of probability and nondeterminism is a case in point; it has received much study [3–5, 8, 9], and the 'right answer' is still by no means settled.

Let us consider the composition of two programs, one probabilistic ($coin\ w$) and one nondeterministic ($arb = True \Box False$). We execute these one after the other, and compare the boolean results:

$$coinarb\ w = coin\ w \gg\!= \lambda c.\,(arb \gg\!= \lambda a.\,(a = c))$$
$$arbcoin\ w = arb \gg\!= \lambda a.\,(coin\ w \gg\!= \lambda c.\,(a = c))$$

It's a two-player game, in which your opponent or fate (*coin*) tosses a coin, and you (*arb*) get to a coin one arbitrarily, and you win if they agree. It clearly makes a difference who plays first! With *coinarb*, fate plays first, and you have complete freedom—to win, to lose, or to leave it up to the coin, or to its opposite; but with *arbcoin*, you play first, and you cannot guarantee to win or to lose. If we stipulate that $\lhd\rhd$ distributes over $\Box$,

$$m \lhd w \rhd (n \Box p) = (m \lhd w \rhd n) \Box (m \lhd w \rhd p)$$

then these consequences come out algebraically:

$$coinarb = coin\ w \Box False \Box True \Box coin\ \bar{w}$$
$$arbcoin = coin\ w \Box coin\ \bar{w}$$

Distributivity means that terms involving both $\Box$ and $\lhd\rhd$ can be normalized into nondeterministic choices of probabilistic choices of values. This suggests that sets of distributions might be a model of the combined theory. However, this turns out not to work [8]. In fact, it is a consequence of idempotence of $\lhd\rhd$ and distributivity over $\Box$ that the semantics must be *convex closed* [1]—if any two distributions m,n are possible outcomes, then so is any convex combination $m \lhd w \rhd n$ of them—and indeed, convex-closed sets of finitely supported distributions do provide a model [3].

Should $\Box$ also distribute over $\lhd\rhd$? It seems not. This does not appear to accord with computational intuitions (although some authors disagree [9]); more importantly, with both directions of distributivity, the distinction between $\Box$ and $\lhd\rhd$ collapses [1].

In previous work [2], we asserted that $\gg\!=$ distributes over $\lhd\rhd$ in both directions—from the right, as above, but also from the left:

$$m \gg\!= \lambda x.\,(k\ x) \lhd w \rhd (l\ x) = (m \gg\!= k) \lhd w \rhd (m \gg\!= l)$$

This is indeed a *theorem* of the model of programs with $\lhd\rhd$ alone, namely probability distributions. But we were wrong to assert it as an *axiom* of $\lhd\rhd$; in conjunction with distributivity of $\lhd\rhd$ over $\Box$, one can conclude the other direction of distributivity, making the theory collapse:

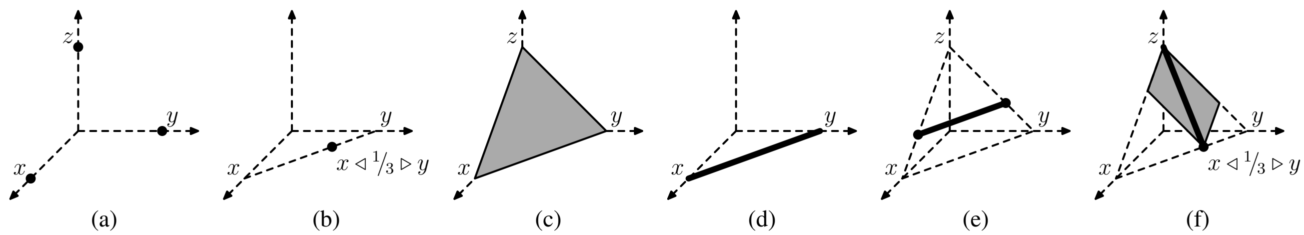

Figure 1. (a) three points x,y,z; (b) point $x \lhd {}^1\!/_3 \rhd y$; (c) triangle xyz; (d) line $x \Box y$; (e) illustrating $(x \Box y) \lhd {}^2\!/_3 \rhd z = (x \lhd {}^2\!/_3 \rhd z) \Box (y \lhd {}^2\!/_3 \rhd z)$; (f) illustrating $(x \lhd {}^1\!/_3 \rhd y) \Box z \neq (x \Box z) \lhd {}^1\!/_3 \rhd (y \Box z)$

$$
\begin{array}{ll}
& m \Box (n \lhd w \rhd p) \\
= & [\![\text{ idempotence }]\!] \\
& (m \lhd w \rhd m) \Box (n \lhd w \rhd p) \\
= & [\![\; m \Box n = arb \ggg\!= \lambda b.\, \mathbf{if}\ b\ \mathbf{then}\ m\ \mathbf{else}\ n \;]\!] \\
& arb \ggg\!= \lambda b.\, \mathbf{if}\ b\ \mathbf{then}\ m \lhd w \rhd m\ \mathbf{else}\ n \lhd w \rhd p \\
= & [\![\text{ promoting } \mathbf{if} \text{ through } \lhd\rhd \;]\!] \\
& arb \ggg\!= \lambda b.\, (\mathbf{if}\ b\ \mathbf{then}\ m\ \mathbf{else}\ n) \lhd w \rhd (\mathbf{if}\ b\ \mathbf{then}\ m\ \mathbf{else}\ p) \\
= & [\![\text{ assuming } \ggg\!= \text{ distributes through } \lhd\rhd \text{ from the left }]\!] \\
& (arb \ggg\!= \lambda b.\, \mathbf{if}\ b\ \mathbf{then}\ m\ \mathbf{else}\ n) \lhd w \rhd (arb \ggg\!= \lambda b.\, \mathbf{if}\ b\ \mathbf{then}\ m\ \mathbf{else}\ p) \\
= & [\![\text{ folding } \Box\text{s back }]\!] \\
& (m \Box n) \lhd w \rhd (m \Box p)
\end{array}
$$

Which is to say, the question of what should be the desirable laws relating $\Box$ and $\lhd\rhd$ is a tricky one. In the remainder of this extended abstract, we present a diagrammatic reasoning technique that might have prevented us from making our earlier mistake.

4. Diagrammatic reasoning

We outline a geometric model, motivated by [3, Chapter 6], of the combined theory of probabilistic and nondeterministic choice from the previous section: idempotence, commutativity, associativity, and algebraicity of $\Box$; identity, idempotence, quasi-commutativity, quasi-associativity, and algebraicity of $\lhd\rhd$; and distributivity of $\lhd\rhd$ over $\Box$. For simplicity, we focus on computations over values x,y,z from a three-valued type. We associate these three points with the basis vectors of a 3-dimensional space, as in Figure 1(a).

A pure computation x always returning the same single value is represented by the single point x in Figure 1(a). A computation exploiting probability but not nondeterminism is still deterministic, so is still represented by a single point, but now a convex combination of the basis vectors. For example, $x \lhd {}^1\!/_3 \rhd y$ is represented by the single point ${}^1\!/_3$ of the way from y to x (that way around), as in Figure 1(b); there is a bijection between distributions over x,y,z and points in the intersection of the plane $x+y+z=1$ with the positive octant (the shaded triangle xyz in Figure 1(c)).

A computation exploiting nondeterminism is represented by a set of points; but as we have seen, idempotence of $\lhd\rhd$ and its distributivity over $\Box$ together imply convex closure, and so it must be a convex set of points. For example, $x \Box y$ is represented by the entire straight line from x to y in Figure 1(d).

In general, computations are represented by convex polygons within the bounded plane xyz. A pure value is represented by one of the three vertices. A nondeterministic choice $m \Box n$ is the convex closure of the union of the polygons m and n. A probabilistic choice $m \lhd w \rhd n$ is the set of all pointwise convex combinations $\{a \lhd w \rhd b \mid a \in m, b \in n\}$ of pairs of points a,b from m and n (which is necessarily convex closed). For composition $m \ggg\!= k$, each vertex a of convex polygon m is a convex combination of the bases x,y,z, and contributes the same convex combination of their images $k\,x, k\,y, k\,z$ under k; then $m \ggg\!= k$ itself is the convex closure of these contributions over all a.

It is not difficult to check that these definitions of $\Box$, $\lhd\rhd$, and $\ggg\!=$ satisfy the individual axioms of nondeterminism and of probability. As for distributivity of $\lhd\rhd$ over $\Box$, consider Figure 1(e). On the left, $x \Box y$ is the base of the triangle (as shown in Figure 1(d)); the convex combination $a \lhd {}^2\!/_3 \rhd z$ of each point a on this base with z yields the line shown in the figure. On the right, $x \lhd {}^2\!/_3 \rhd z$ and $y \lhd {}^2\!/_3 \rhd z$ are the two points in the figure, and their convex closure is the same line as on the left. A similar argument holds for any weight and any convex polygons, not just for values.

Conversely, consider Figure 1(f), which illustrates the undesirable opposite distributivity. On the left, the point $x \lhd {}^1\!/_3 \rhd y$ is shown on the base of the triangle, and the line is the $\Box$ of that and z. On the right, $x \Box z$ and $y \Box z$ are the two upper edges of the triangle, and their pointwise convex combination is the entire shaded parallelogram; these two are clearly not equal. Intuitively, this law would duplicate a nondeterministic choice, expanding the possible outcomes.

5. Conclusion

As discussed, combining probability and nondeterminism in a functional program is by no means a straightforward endeavour. The model of convex sets of distributions does, however, admit a pleasing diagrammatic interpretation of such computations. Diagrammatic arguments, consisting of points, lines and convex polygons, offer a quick visual method for verifying (or disproving) program equations. As such, they form a useful complementary technique to add to the reasoning programmer's arsenal.

References

[1] J. Gibbons. Unifying theories of programming with monads. In *UTP Symposium*, volume 7681 of *LNCS*, pages 23–67. Springer, 2013.

[2] J. Gibbons and R. Hinze. Just do it: Simple monadic equational reasoning. In *ICFP*, pages 2–14, Sept. 2011.

[3] A. McIver and C. Morgan. *Abstraction, Refinement and Proof for Probabilistic Systems*. Springer Verlag, 2005.

[4] L. Meinicke and K. Solin. Refinement algebra for probabilistic programs. *FAC*, 22:3–31, 2010.

[5] M. W. Mislove. Nondeterminism and probabilistic choice: Obeying the laws. In C. Palamidessi, editor, *CONCUR*, volume 1877 of *LNCS*, pages 350–364. Springer, 2000.

[6] G. D. Plotkin and J. Power. Notions of computation determine monads. In *FOSSACS*, volume 2303 of *LNCS*, pages 342–356. Springer, 2002.

[7] G. D. Plotkin and M. Pretnar. Handlers of algebraic effects. In *ESOP*, volume 5502 of *LNCS*, pages 80–94. Springer, 2009.

[8] D. Varacca and G. Winskel. Distributing probability over nondeterminism. *MSCS*, 16(1):87–113, 2006.

[9] W. Yi and K. G. Larsen. Testing probabilistic and nondeterministic processes. In *Protocol Specification, Testing and Verification XII*, pages 47–61, 1992.