
ELSEVIER

Theoretical Computer Science 266 (2001) 513–541

Theoretical Computer Science

www.elsevier.com/locate/tcs

Partial correctness for probabilistic demonic programs

A.K. McIver *, Carroll Morgan

Computing Laboratory, Programming Research Group, Oxford University, Wolfson Building, Parks Road, Oxford OX1 3QD, UK

Received December 1997; revised April 2000; accepted May 2000
Communicated by G. Plotkin

Abstract

Recent work in sequential program semantics has produced both an operational (He et al., Sci. Comput. Programming 28(2, 3) (1997) 171–192) and an axiomatic (Morgan et al., ACM Trans. Programming Languages Systems 18(3) (1996) 325–353; Seidel et al., Tech Report PRG-TR-6-96, Programming Research group, February 1996) treatment of total correctness for probabilistic demonic programs, extending Kozen's original work (J. Comput. System Sci. 22 (1981) 328–350; Kozen, Proc. 15th ACM Symp. on Theory of Computing, ACM, New York, 1983) by adding demonic nondeterminism. For practical applications (e.g. combining loop invariants with termination constraints) it is important to retain the traditional distinction between partial and total correctness. Jones (Monograph ECS-LFCS-90-105, Ph.D. Thesis, Edinburgh University, Edinburgh, UK, 1990) defines probabilistic partial correctness for probabilistic, but again not demonic programs. In this paper we combine all the above, giving an operational and axiomatic framework for both partial and total correctness of probabilistic and demonic sequential programs; among other things, that provides the theory to support our earlier – and practical – publication on probabilistic demonic loops (Morgan, in: Jifeng et al. (Eds.), Proc. BCS-FACS Seventh Refinement Workshop, Workshops in Computing, Springer, Berlin, 1996).

Keywords: Program logic; Verification; Probability; Partial correctness

1. Introduction

An *operational* model provides a concrete description of realistic program behaviour; on the other hand, *program logic* is more suited to validation. Thus, a compelling aim is to encapsulate a plausible operational model as an interpretation of a system of logical axioms. In this paper we treat these two themes for probabilistic programming.

* Corresponding author.
E-mail addresses: anabel@comlab.ox.ac.uk (A.K. McIver), carroll@comlab.ox.ac.uk (C. Morgan).

0304-3975/01/$ - see front matter
PII: S0304-3975(00)00208-5

It has long been understood that *demonic nondeterminism*, the mathematical notion encapsulating 'abstraction', is vital for retaining simplicity in reasoning and expression [3]. The introduction of *probability*, a special but distinct case of (demonic) nondeterminism, leads to some unexpected consequences [20], and this observation together with the realisation that the distinction is (among other things) necessary for the realistic modelling of many probabilistic distributed algorithms suggests that instead of discarding one in favour of the other, rather both should coexist and be understood in any useful model. (Segala gives a nice exposition of these issues [24].) The wish to understand nontermination needs no explanation.

Thus our first contribution is theoretical: it is a correspondence between operational and logical descriptions of sequential, probabilistic, demonically nondeterministic and possibly nonterminating programs. This result is achieved by using a 'quantitative logic' whose expressions represent probabilistic rather than absolute judgements concerning program correctness.

Others have studied logic and probability besides ourselves, but since nondeterminism, probability and nontermination together pose a real challenge [7, p. 200], one finds simpler, more idealistic situations in the literature. Either the logics are not quantitative (thus only events with definite 0 or 1 probabilities can be analysed) [23], or of those that do allow quantitative judgements none treat (in addition) both nontermination and nondeterminism [20, 7, 2]. The novelty here is that we account for all three.

A second contribution is practical: it is a justification, using our program logic, of invariant/variant principles based on *wp* and *wlp* [3] for probabilistic demonic loops, thus (finally) setting the validation of small, probabilistic programs on a par with standard methods.

Crucial in this are the axioms of the quantitative logic that characterise feasible program behaviour. Their discovery relies on our theoretical analysis, and yet they provide the key for the nonobvious yet sound (compositional) probabilistic judgements in nondeterministic and (possibly) nonterminating environments.

This beautiful connection between operational models and quantitative logic, allowing the developments here of simple proof rules, was originally revealed in the early 1980s by Kozen [14] and subsequently by Jones [2] both of whom broke new ground by showing that pure probabilistic computations could be explained using standard domain theoretical constructions applied to an underlying domain containing probability distributions. Later Morgan et al. [20] extended that idea to a more complicated powerdomain, successfully combining probabilistic and demonic nondeterministic behaviour; in doing so they showed that the generalisation to real-valued expectations is (unlike in Kozen's and Jones' pure probabilistic setting) fundamental for retaining compositionality [19]. The final (missing) ingredient – nontermination – can be added similarly by using the still more complicated Plotkin construction; we set out that construction here in Section 2.

But our practical goal – a sound justification of the separation of correctness and termination (Sections 4 and 5) – can only be demonstrated with this construction provided that the resulting Plotkin powerdomain decomposes into the corresponding Smyth

and Hoare powerdomains, for they model, respectively, terminating and nonterminating behaviour. General results from domain theory guarantee that vital decomposition for certain kinds of underlying powerdomain, and the pleasant surprise is that both discrete and continuous probability distributions are amenable specialisations – although we must use a nonstandard domain for the continuous case [4].

Our tendency though is to favour the discrete distributions, and not only because of their marked simplicity when compared with the continuous case. Most published probabilistic algorithms only generate discrete distributions, and hence a 'discrete theory' is almost always sufficient; but more generally still, one could argue (as Kozen does [13]) that probabilistic *computations* themselves are essentially discrete in nature. Thus the construction for the continuous case is sketched in a separate section (Section 6).

An extensive discussion of examples, and the general treatment of loops, is given elsewhere [17].

Throughout we use infix dot '.' for function application, associating to the left so that $f.x.y$ means $(f(x))(y)$; and we write ':=' for 'is defined to be equal to'.

2. A convex powerdomain of distributions

In this section we consider some general results of powerdomains specialised to a domain of probability distributions.

In program semantics, powerdomains are used to study nondeterminism, a phenomenon arising when a program might output any one of some set of results rather than a single, determined function of its input. The details of any powerdomain are proscribed by the way it orders those result sets, and the particular choice of order depends on criteria which can be explained in terms of the desired treatment of programs' possible nonterminating behaviour. All, however, regard nondeterminism as demonic, and thus they provide the starting point for extending probability.

The Symth order[1] (Definition B3) treats nontermination as the worst behaviour and thus the Smyth powerdomain models total correctness. Similarly, the Hoare order (Definition B4) models partial correctness: nontermination is treated as the best outcome in that order. The Plotkin powerdomain (Definition B2) uses the Egli–Milner order (Definition B5) and combines both views; thus that is what we shall use in our operational semantics.

Recursion is an obvious 'source' of nonterminating behaviour, and as usual we shall model it as a least fixed point (with respect to the Egli–Milner order); a principal concern therefore is to ensure that any Egli–Milner limit can be recovered as a Smyth limit and Hoare limit separately, for only then can the Egli–Milner order alone be used to encode the other two, and thus provide a basis for the sound separation of

[1] For this and other facts and definitions from domain theory, we follow the conventions set out in [1]. We summarise the details for this paper (often specialising them to our particular application) in Appendix B.

correctness reasoning into partial and total. If such is the case, we say that the Plotkin powerdomain is decomposable (into the Smyth and Hoare powerdomains).

In general, the Plotkin powerdomain is not decomposable, but in some special cases it is: Abramsky and Jung [1] show that one such case is when the underlying domain is ω-continuous (Definition B10), and the main result of this section is to exhibit this concretely for a domain of probability distributions.

We begin by summarising some consequences of Abramsky and Jung's results – specifically they prove an isomorphism [1] (reproduced here in Theorem B.1) between the abstract Plotkin powerdomain over an ω-continuous complete partial order and the space of lenses. We write $(D, \leqslant)$ for a (general) ω-continuous complete partial order, and $(Lens(D), \leqslant_{TEM})$ (Definition B13) for its associated space of lenses. $Lens(D)$ is important because it provides a suitable powerdomain of Egli–Milner closed sets (Definition B2) of D, whilst the isomorphism provides us with decomposition results on that powerdomain.

In general, a set is Egli–Milner closed if it is the intersection of an *up-closed* (*Smyth-closed*) set (Definition B1) and a *down-closed* (*Hoare-closed*) set (Definition B1). A subset of D is contained in $Lens(D)$ if it is the intersection of a Scott-compact (Definition B12) up-closed set and a Scott-closed (Definition B11) (hence down-closed) set. Together the conditions imply that elements in $Lens(D)$ are Scott-compact, and in any case are Egli–Milner-closed. The additional closure conditions will provide us with our decomposition results.

Next, we describe the two corollaries of the general isomorphism Theorem B.1 which imply that even directed limits of elements in $Lens(D)$ can be decomposed into two sets, one representing the Smyth limit and the other the Hoare limit separately. We write $\sqcup_{EM}$, $\sqcup_S$ and $\sqcup_H$ for, respectively, the Egli–Milner, Smyth and Hoare limits. For a subset A of D we write $sc.A$ for the smallest Scott-closed set containing A (Definition B11).

Corollary 2.1. *For any $\leqslant_{TEM}$-directed subset $\mathscr{A}$ of $Lens(D)$ the limit $\sqcup_{TEM}\mathscr{A}$ exists, and satisfies*

$$\sqcup_{TEM}\mathscr{A} = \sqcup_S\mathscr{A} \cap sc.(\sqcup_H\mathscr{A}).$$

(Insisting on closure after $\sqcup_H$ can be seen as a continuity condition,[2] and in any case it selects the least lens greater than all those in $\mathscr{A}$.)

Proof. The lemma is a consequence of the isomorphism between the abstract Plotkin powerdomain (Definition B15) and the space $(Lens(D), \leqslant_{TEM})$. Both the limit and its

[2] Consider the $\leqslant_{TEM}$ chain on sets of real intervals in $[0,1]$,

$$\{0\} \leqslant_{TEM} \{1/2\} \leqslant_{TEM} \cdots \leqslant_{TEM} \{(n-1)/n\} \leqslant_{TEM} \cdots, \tag{1}$$

which has limit $\{1\}$ (the limit point of the underlying series). The union of the down sets is the half-closed interval $[0,1)$, but the intersection of the up sets is $\{1\}$. Failing to limit-close the Hoare limit would produce an empty result. Nevertheless, $\{1\}$ is the least lens in $[0,1]$ greater than all the lenses in the chain.

decomposition (1) exist in the abstract powerdomain in general [12], and the isomorphism is given by Abramsky and Jung [1] and is reproduced here in Theorem B.1. □

The next result shows that the $\sqcup_{TEM}$ limit determines the Smyth limit (in the Smyth ordering) and the Hoare limit (in the Hoare ordering). We write $\equiv_S$, $\equiv_H$, respectively, for Smyth equivalence and Hoare equivalence (Definition B17) between elements in $Lens(D)$: our lemma below shows in addition that the limits are indistinguishable relative to the appropriate equivalences.

Corollary 2.2. *For any $\leqslant_{EM}$-directed subset $\mathscr{A}$ of $Lens(D)$, the following equivalences hold*:

$$\sqcup_{TEM}\mathscr{A} \equiv_S \sqcup_S \mathscr{A}$$

$$\sqcup_{TEM}\mathscr{A} \equiv_H \sqcup_H \mathscr{A}.$$

Proof. This too is a property of abstract Plotkin powerdomains [12], and so follows from the isomorphism (Theorem B.1) used in the proof of Corollary 2.1. □

We now turn specifically to probabilistic semantics: our task is to exploit the general result Corollary 2.2 to a domain of probability distributions, and for that we need only show that our space of interest is ω-continuous.

We write S for the state space and assume (for now) that it is countable. The space of (discrete) probability distributions[3] over S is defined as follows.

Definition 2.3. For state space S, the space of *distributions* $(\bar{S}, \sqsubseteq)$ over S is defined

$$\bar{S} := \left\{ F \mid S \to [0,1]; \quad \sum_{s:S} F.s \leqslant 1 \right\}$$

and for F, F' in $\bar{S}$ we define the order

$$F \sqsubseteq F' := (\forall s: S)(F.s \leqslant F'.s).$$

These special distributions are more precisely called *discrete sub-probability measures* [11]; they do not necessarily sum to 1, and the deficit gives the probability of nontermination. The 'everywhere zero' distribution for example, that assigns zero probability to all states, models nowhere-terminating behaviour. (An alternative though less convenient treatment would assign probability 1 to some special state $\perp$.)

Now, we show that $(\bar{S}, \sqsubseteq)$ is an ω-continuous complete partial order.

[3] The more general notion is that of 'valuation' over the Scott topology of a partially ordered set (see Section 6). For discrete distributions over a countable, flat domin S, the space of valuations reduces to the ordinary definition of discrete probability distributions; thus for the present we persist with that terminology.

Lemma 2.4. *For a countable state space S, its distributions $(\bar{S}, \sqsubseteq)$ form an ω-continuous complete partial order.*

Proof. The completeness of $(\bar{S}, \sqsubseteq)$ is trivial, given the completeness of the interval $[0,1]$ under $\leqslant$ over the reals.

To show that $\bar{S}$ is ω-continuous we need only exhibit a countable basis (Definition B9). One such is the set of distributions contained in

$$\{F \downarrow T \mid F : S \to ([0,1] \cap \mathbb{Q});\ T \in \mathbb{P}_{Fin} S\}, \tag{2}$$

where $F \downarrow T$ is the function equal to F on T, and to zero outside of T, and $\mathbb{P}_{Fin} S$ is the set of *finite* subsets of S. Since S is countable, so is (2), and moreover since any real is the least upper bound of rationals way-below it (Definition B8), we have a basis. □

Lemma 2.4 shows that $(Lens(\bar{S}), \sqsubseteq_{TEM})$ satisfies conditions necessary for the decomposition of Corollary 2.2, but that space is only relevant in our context provided the order $\sqsubseteq_{TEM}$ between lenses reduces to the ordinary Egli–Milner order $\sqsubseteq_{EM}$ since the latter is what we use in program semantics. The next lemma shows that to be the case.

Lemma 2.5. *If $A, A' \subseteq Lens(\bar{S})$ then*

$$A \sqsubseteq_{TEM} A' \quad \textit{iff} \quad A \sqsubseteq_{EM} A'.$$

Proof. We show for any lens A that $\downarrow A = sc.(\downarrow A)$, for (from Definitions B16 and B5) that is sufficient to imply correspondence of the orders. First, we note that $\downarrow A \subseteq sc.(\downarrow A)$, thus we shall concentrate on the alternative inclusion. Let the limit a in $sc.(\downarrow A)$ be generated by the chain $a_0 \sqsubseteq a_1 \sqsubseteq a_2 \sqsubseteq \cdots$, where $a_i \in \downarrow A$ for all i. The result follows provided that $a \in \downarrow A$, or equivalently if $A \cap \uparrow\{a\} \neq \emptyset$. We reason as follows:

$$\begin{aligned}
& A \cap \uparrow \{a\} \\
&= A \cap \bigcap_{i \geqslant 0} \uparrow \{a_i\} \quad a \text{ is the limit} \\
&= \bigcap_{i \geqslant 0} A \cap \uparrow \{a_i\} \\
&\neq \emptyset \quad A \cap \uparrow \{a_i\} \neq \emptyset;\ \text{see below.}
\end{aligned}$$

For the deferred justification, we note first that the sets $A \cap \uparrow\{a_i\}$ form a chain (with respect to reverse subset inclusion) of nonempty, compact sets: each set $A \cap \uparrow\{a_i\}$ is compact because it is an intersection of two compact sets (A is a lens, thus is compact, whereas $\uparrow\{a_i\}$ is the up-closure of a singleton, also compact), and in $\bar{S}$ the intersection of compact sets is compact. (Compact, up-closed sets are the intersection of finitary hyperspaces described in Appendix A; and it can be shown that arbitrary intersections

of such hyperspaces are again compact.) Nonemptiness of $\bigcap_{i\geqslant 0} A \cap \uparrow\{a_i\}$ now follows since the set of nonempty compact subsets (of an ω-continuous domain) ordered by reverse inclusion is a complete partial order [1, p. 61]. □

We are now ready to define a Plotkin-style powerdomain for probability distributions. We select a subset of $Lens(\bar{S})$ as follows by imposing the further closure condition of '(probabilistic) convexity' (defined below) (because in our application to probability, taking the whole of $Lens(\bar{S})$ is still not suitable for probabilistic program semantics). For distributions F, F' in $\bar{S}$ and p in $[0,1]$ we can form $F_p\oplus F'$, the weighted average, defined pointwise over S as $p\times F+(1-p)\times F'$ (with usual scalar multiplication and addition).

Definition 2.6. For p in $[0,1]$ and subsets A, A' of $\bar{S}$ we define

$$A_p\oplus A' := \{F_p\oplus F' \mid F : A, F' : A'\}.$$

We say that A is (probabilistically) *convex* if $A_p\oplus A = A$ for all p in $[0,1]$.

Our convexity condition – the only novel closure condition in this context, but one we have used elsewhere [20, 9] – ensures (among other things) that, in a programming context, nondeterministic choice can always be 'refined by' probabilistic choice. Other laws between program operators also hold because of the convex condition, and a full description of them can be found elsewhere [9].

Our probabilistic powerdomain over $\bar{S}$ is defined next.

Definition 2.7. The *convex powerdomain* $(\mathscr{C}S, \sqsubseteq_{EM})$ over the space of distributions $\bar{S}$ comprises the (probabilistically) convex sets in $Lens(\bar{S})$. Its order $\sqsubseteq_{EM}$ is the usual Egli–Milner order (Definition B5).

Our final task for this section is to show that the decomposition results Corollaries 2.1 and 2.2 apply even within $\mathscr{C}S$. That at last gives us the main result of this section: Egli–Milner limits in $\mathscr{C}S$ determine separately the Smyth and Hoare limits.

Theorem 2.8. *For any $\sqsubseteq_{EM}$-directed subset $\mathscr{A}$ of $\mathscr{C}S$, the following equivalences hold*:

$$\sqcup_{EM}\mathscr{A} \equiv_S \sqcup_S \mathscr{A}$$

$$\sqcup_{EM}\mathscr{A} \equiv_H sc.(\sqcup_H \mathscr{A}).$$

Proof. Given Lemmas 2.4 and 2.5, the result follows immediately from Corollaries 2.1 and 2.2 provided the additional closure condition on $\mathscr{C}S$, namely convexity, holds of $\sqcup_S\mathscr{A}\cap sc.(\sqcup_H\mathscr{A})$ in the case that all the elements of $\mathscr{A}$ themselves are convex. That follows from these elementary facts: up-closing preserves convexity ($\sqsubseteq$-monotonicity of ${}_p\oplus$); the intersection of convex sets is convex; down-closing preserves convexity

(similar to up-closing); the union of a $\subseteq$-directed set of sets is convex; and limit-closing preserves convexity ($\sqsubseteq$-continuity of ${}_p\oplus$). $\square$

This section has defined the convex powerdomain, whose use for modelling probabilistic imperative programs now follows from the constructions for the Smyth-style domain [9]: for example the sequential composition is a generalised functional composition; nondeterministic choice is union (then convex closure); and the probabilistic choice is weighted average as defined above. In Section 3 we give further details.

For recursion one takes limits of chains, and here is the significance of Theorem 2.8: we must be sure that taking the limit in the convex domain agrees with the more specialised limit in the Smyth domain *and* the Hoare domain – for that is what allows us to use the more general convex domain for either. It is known that the equivalence holds for standard (nonprobabilistic) domains; Theorem 2.8 confirms the preservation of the property when probability is included.

Now, we turn to programs and logic.

3. Probabilistic programs and logic

The results of the last section have provided the tools for an operational model, which (via the Egli–Milner order) captures the essence of both termination and nontermination. We now consider how to characterise that model using axioms of a quantitative logic, beginning with a review of traditional methods.

Over standard (nonprobabilistic) demonic programs, a popular model for total correctness is $S \to \mathscr{S}S_\perp$, where $S_\perp$ is the flat domain extending state space S with $\perp$ for nontermination, and $\mathscr{S}$ forms the Smyth powerdomain over that; Dijkstra's weakest 'ordinary' preconditions $\mathbb{P}S \to \mathbb{P}S$ [3] support a programming logic suitable for total correctness. For partial correctness one can use $S \to \mathscr{H}S_\perp$ (Hoare) for the model and weakest 'liberal' preconditions for the logic. Finally, although partial and total correctness are available simultaneously via $S \to \mathscr{G}S_\perp$ (Plotkin), for r in $S \to \mathscr{G}S_\perp$ and postcondition Q in $\mathbb{P}S$ still it is more convenient to define separately

$$\begin{aligned} wp.r.Q &:= \{s \mid r.s \subseteq Q\} \quad \text{weakest precondition} \\ wlp.r.Q &:= \{s \mid r.s \subseteq Q_\perp\} \quad \text{weakest liberal precondition} \end{aligned} \tag{3}$$

to give the total (wp) and partial (wlp) programming logics. Note that the definitions (3) work together only over $\mathscr{G}S_\perp$ (the intersection of $\mathscr{H}S_\perp$ and $\mathscr{S}S_\perp$) – wp does not work over $\mathscr{H}S_\perp$ and wlp does not work over $\mathscr{S}S_\perp$. (Nelson [21] gives a nice treatment of the issues.)

For probabilistic programs, He et al. [9] propose $S \to \mathscr{C}_S S$ for total correctness, where $\mathscr{C}_S S$ is convex like $\mathscr{C}S$ of the previous section, but based on the Smyth order. Morgan et al. [20] provide a probabilistic 'greatest pre-expectation' logic for that, where *expec-*

tations are nonnegative real-valued functions over the state space (extending Kozen's treatment [14] for nondemonic programs).

To access total and partial correctness simultaneously, by analogy with the standard case we replace He's Smyth-based $\mathscr{C}_S S$ by our more sophisticated Egli–Milner-based $\mathscr{C}S$. From there we could go on immediately to generalise *wp* and *wlp* separately (as at (3) above), but we do not do so. Instead we allow expectations to range over *negative* as well as nonnegative values: we define *ewp*, the operator underlying the other two logics, which exactly characterises our operational model and from which they can be extracted. Roughly speaking, total correctness results are obtained from nonnegative postexpectations and partial correctness results from nonpositive. That we can unify partial and total correctness with a single expectation transformer speaks of the greater expressivity of numbers when compared with the booleans.

We begin the details with the construction of the operational model for probabilistic, demonic model of programs.

Definition 3.1. For a countable state space S the space of (*discrete*) *probabilistic*, *demonic programs* $(\mathscr{M}S, \sqsubseteq_{EM})$ is given by

$$\mathscr{M}S := S \to \mathscr{C}S$$

with the order induced pointwise from $\mathscr{C}S$, so that for r, r' in $\mathscr{M}S$ we define

$$r \sqsubseteq_{EM} r' := (\forall s\colon S)(r.s \sqsubseteq_{EM} r'.s).$$

We occasionally use $\sqsubseteq_S$ and $\sqsubseteq_H$ over $\mathscr{M}S$, analogously lifted from $\mathscr{C}S$.

Thus, our programs take initial states to sets of final distributions: the plurality of the sets represents *demonic* nondeterminism; the distributions they contain each represent *probabilistic* nondeterminism.

The next task is to investigate the dual representation of programs as expectation transformers. We extend the expectations found in [20, 17], where the topic was total correctness (the Smyth order and up-closed sets) and expectations were of type $S \to [0,1]$, by using $[-1,1]$ instead: we write $\mathscr{E}S$ for $S \to [-1,1]$, and use lower-case Greek letters for typical elements.[4]

Expectation transformers $\mathscr{T}S$ are thus functions of type $\mathscr{E}S \to \mathscr{E}S$. We write $\int_F \alpha$ for the expected value of α in $\mathscr{E}S$ averaged over distribution F in $\bar{S}$.[5] As a special case of expectations, we interpret predicates as $\{0,1\}$-valued functions of the state space, and for predicate A holding at state s we write either $s \in A$ or $A.s = 1$ as convenient. For a scalar c we write $\underline{c}$ for the constant expectation evaluating to c over all of S.

[4] We restrict expectations to the interval $[-1,1]$ because it is more convenient for our application to partial correctness. However, for the general program logic Definition 3.2, the restriction is only apparent since those transformers are scaling (a consequence of Definition 3.8) implying that by suitably scaling the post condition to lie in that range, the effect of program behaviour on bounded functions generally can be determined by functions over $[-1,1]$.

[5] The expected value for a discrete distribution F is actually given by $\sum_{s:S} \alpha.s \times F.s$.

With those conventions the predicates *true* and $false$ correspond to the expectations $\underline{1}$ and $\underline{0}$ respectively. Finally, for relations between expectations we write

$\Rrightarrow$ – everywhere no more than

$\equiv$ – everywhere equal to

$\Lleftarrow$ – everywhere no less than,

so that we generalise, respectively, implication, equivalence and reverse implication on predicates.[6] Our logic is based on the 'extended greatest pre-expectation transformer', defined as follows.

Definition 3.2. Let r be a program in $\mathscr{M}S$, taking initial states in S to sets of final distributions over S. Then the *greatest* pre-expectation at state s of program r, with respect to post-expectation α in $\mathscr{E}S$, is defined

$$ewp.r.\alpha.s := \sqcap \left\{ \int_F \alpha \mid F{:}r.s \right\}.$$

The effect of the definition is to consider all possible post-distributions F in $r.s$, and then demonically to choose the one that gives the least (the 'worst') expectation for the post-expectation α: thus nondeterminism is demonic in that it minimises the pre-expectation at each initial state, and Definition 3.2 is then the greatest expectation everywhere no more than those pointwise minima.

For standard programs, if executing a program r from a state s is *certain* to establish a postcondition A then that state is contained in the associated weakest precondition; with our definition we would have $ewp.r.A.s = 1$. For probabilistic programs, if the standard postcondition A is established with only a probability at least p say, then the greatest preexpectation on executing r from s initially is at least p and we have $ewp.r.A.s = p$.[7] Thus as a special case, when A is a predicate we can interpret $ewp.r.A.s$ as the greatest assured probability that A holds after execution of r from s.

Now, we discover the various refinement orders over $\mathscr{T}S$ that correspond via *ewp* with orders over the operational $\mathscr{M}S$. First, we generalise the observation from standard programming (e.g. [21]) that the Smyth order on programs corresponds to the implication order lifted to predicate transformers and that the Hoare order similarly corresponds to (lifted) reverse implication. We use $\mathscr{P}S$ (typical element π) to denote the set of nonnegative valued expectations and $\mathscr{N}S$ (typical element ν) for the nonpositive valued expectations. They are both subsets of $\mathscr{E}S$.

[6] Although the order is just $\leqslant$ lifted to functions, we prefer to use $\Rrightarrow$ because of its similarity to $\Rightarrow$ of ordinary Boolean-based logic.

[7] The apparent confusion between expectations and probabilities is deliberate and harmless: the probability of an event A over a distribution is equal to the expected value of (the characteristic function of) A over that same distribution.

Lemma 3.3. *For r, r' in $\mathscr{M}S$, and expectations π in $\mathscr{P}S$ and v in $\mathscr{N}S$,*

$$r \sqsubseteq_S r' \text{ implies } ewp.r.\pi \Rrightarrow ewp.r'.\pi$$

$$r \sqsubseteq_H r' \text{ implies } ewp.r.v \Lleftarrow ewp.r'.v.$$

Proof. For r, r' in $\mathscr{M}S$, any s in S and π in $\mathscr{P}S$ we reason as follows:

$$r \sqsubseteq_S r'$$

implies $\uparrow(r.s) \supseteq r'.s$ definition $\sqsubseteq_S$

implies $\sqcap\left\{\int_F \pi \mid F : \uparrow(r.s)\right\} \leqslant \sqcap\left\{\int_F \pi \mid F : \uparrow(r'.s)\right\}$

iff $\sqcap\left\{\int_F \pi \mid F : (r.s)\right\} \leqslant \sqcap\left\{\int_F \pi \mid F : (r'.s)\right\}$ $\underline{0} \Rrightarrow \pi$; see below

iff $ewp.r.\pi.s \leqslant ewp.r'.\pi.s$. Definition 3.2.

For the deferred justification we appeal to the monotonicity of the arithmetic over nonnegative arguments without subtraction: $r.s$ differs from $\uparrow(r.s)$ only by the addition of 'larger elements' according to Definition 2.3, and the minimum selection on the left cannot be *increased* by removing the up-closure.

The result now follows by generalising on s, and a similar argument justifies the second statement (but note the reversal $\Lleftarrow$). □

Lemma 3.3 is the key to defining the expectation-transformer equivalents to the Smyth, Hoare and Egli–Milner orders where, as usual, the Egli–Milner order is the intersection of the Smyth and Hoare orders.

Definition 3.4. For t, t' in $\mathscr{T}S$ we define

$$t \sqsubseteq_S t' := (\forall \pi : \mathscr{P}S)(t.\pi \Rrightarrow t'.\pi),$$

$$t \sqsubseteq_H t' := (\forall v : \mathscr{N}S)(t.v \Lleftarrow t'.v),$$

$$t \sqsubseteq_{EM} t' := t \sqsubseteq_S t' \wedge t \sqsubseteq_H t'.$$

That the Egli–Milner order between programs is preserved under *ewp* now follows directly.

Corollary 3.5. *For r, r' in $\mathscr{M}S$,*

$$r \sqsubseteq_{EM} r' \text{ implies } ewp.r \sqsubseteq_{EM} ewp.r'.$$

Proof. Lemma 3.3 and Definition 3.4. □

The corollary shows only that *ewp* is an order-preserving mapping between $(\mathscr{M}S, \sqsubseteq_{EM})$ and $(\mathscr{T}S, \sqsubseteq_{EM})$. The next result, the converse of Corollary 3.5, shows that it is

also an injection, and therefore that programs can be modelled equivalently either as relations or as expectation transformers.

Lemma 3.6. *For r, r' in $\mathscr{M}S$, if $ewp.r \sqsubseteq_{EM} ewp.r'$ then $r \sqsubseteq_{EM} r'$.*

Proof. Suppose for a contradiction that $ewp.r \sqsubseteq_{EM} ewp.r'$ but $r \not\sqsubseteq_{EM} r'$, for some r, r' in $\mathscr{M}S$. Assume first that $r \not\sqsubseteq_S r'$, so that for some distribution F and state s we have both

$$F \notin \uparrow (r.s) \tag{4}$$

and

$$F \in r'.s. \tag{5}$$

From (4) and since $\uparrow(r.s)$ is compact, with the aid of Lemma A.2 we have for some expectation π in $\mathscr{P}S$ that

$$\int_F \pi \;<\; \sqcap \left\{ \int_{F'} \pi \mid F' : \uparrow(r.s) \right\}$$

and thus that $\int_F \pi < ewp.r.\pi.s$. From (5) however we have $ewp.r'.\pi.s \leqslant \int_F \pi$ directly, giving together

$$ewp.r'.\pi.s \leqslant \int_F \pi \;<\; ewp.r.\pi.s$$

and contradicting the hypothesis (at the state s).

The alternative, that $r \not\sqsubseteq_H r'$ is treated similarly, but appealing to Lemma A.3. □

Since in the proof of Lemma 3.6 we have actually proved the separate converses to Lemma 3.3, we can now state the correspondence between the relational model and program logic for all three orders.

Theorem 3.7. *The following equivalences hold for all r, r' in $\mathscr{M}S$:*

$$r \sqsubseteq_S r' \quad \text{iff} \quad ewp.r \sqsubseteq_S ewp.r',$$

$$r \sqsubseteq_H r' \quad \text{iff} \quad ewp.r \sqsubseteq_H ewp.r',$$

$$r \sqsubseteq_{EM} r' \quad \text{iff} \quad ewp.r \sqsubseteq_{EM} ewp.r'.$$

We have thus shown that *ewp* order-embeds $\mathscr{M}S$ into $\mathscr{T}S$.

But there are many $\sqsubseteq_{EM}$-monotonic expectation transformers that are not *ewp*-images of $\mathscr{M}S$. The final result of this section completes our exact logical characterisation of the convex powerdomain: we identify 'healthiness conditions' over $\mathscr{T}S$ in the style of Dijkstra [3] (for standard programs) and of Morgan et al. [20] (for probabilistic programs) that distinguish (images through *ewp* of) programs of $\mathscr{M}S$ within it. The

importance of the result is that theorems proved within $\mathscr{T}S$ about healthy expectation transformers correspond to theorems about programs in $\mathscr{M}S$.

The first healthiness condition is a slight generalisation of the sublinearity of Morgan [20]. To state it we define, for expectations α, β in $\mathscr{E}S$ and real nonnegative scalar c, the expectations $\alpha + \beta$ and $c\alpha$, where (as for p-averaging of distributions) we mean a pointwise lifting of standard addition and scalar multiplication.

Definition 3.8. An expectation transformer t in $\mathscr{T}S$ is *sublinear* iff, for all α, β in $\mathscr{E}S$, and a, b, c nonnegative reals,

$$t.(a\alpha + b\beta - \underline{c}) \Leftarrow a(t.\alpha) + b(t.\beta) - \underline{c}.$$

A second condition is *bounded continuity*: transformers satisfy bounded continuity provided they distribute up- (down-) directed limits in $\mathscr{P}S$ ($\mathscr{N}S$).

We note first that both sublinearity and bounded continuity are satisfied by all images of $\mathscr{M}S$ under *ewp*.

Lemma 3.9. *Any expectation transformer ewp.r, for r in $\mathscr{M}S$, is sublinear and boundedly continuous.*

Proof. Definition 3.2, compactness of programs' result sets and properties of arithmetic ([20, 15] give more detailed proofs). □

For total correctness (for the Smyth $\mathscr{C}_S$), sublinearity and bounded continuity tell the whole story [20, 15, Theorem 8.7]; in our more general $\mathscr{C}S$ however, there are sublinear elements of $\mathscr{M}S$ that are not *ewp*-images. Take for example S to be the two-element state space $\{x, y\}$, and consider the result set

$$\{F : \bar{S} \mid F.x = F.y\}.$$

It is convex, but not Egli–Milner closed;[8] its associated expectation transformer formed by *ewp* is sublinear, but it is not the *ewp*-image of any element of $\mathscr{M}S$.

The characterisation of Egli–Milner closure is captured by a second healthiness condition – 'partial linearity' – which states that $t.\alpha$ depends only on the pre-expectations of t applied to expectations in $\mathscr{P}S \cup \mathscr{N}S$.

Definition 3.10. An expectation transformer, t in $\mathscr{T}S$ is said to be *partially linear* if for all states s in S, and all expectations α in $\mathscr{E}S$ which are zero on all but a finite subset of S, there are expectations π in $\mathscr{P}S$ and ν in $\mathscr{N}S$ such that $\alpha = \pi + \nu$ and

$$t.\alpha.s = t.\pi.s + t.\nu.s.$$

[8] In fact, its closure is $\{F : \bar{S} \mid F.x, F.y \leqslant \frac{1}{2}\}$, from which it is indistinguishable using *ewp* for any α in $\mathscr{P}S \cup \mathscr{N}S$.

Note that the implicit existential quantification in Definition 3.10 means there may be many decompositions of α as a sum $\pi + v$.[9]

We complete the correspondence between healthy expectation transformers and $\mathscr{M}S$ with the next theorem, which we state only. The proof is omitted as it is overly technical and not necessary for the rest of the paper.

Theorem 3.11. *An expectation transformer t in $\mathscr{T}S$ is boundedly continuous, sublinear and partially linear if and only if there is r in $\mathscr{M}S$ such that $t = ewp.r$.*

Theorem 3.11 concludes our logical characterisation of our convex powerdomain. In the next section we turn to applications, and discover that the healthiness conditions of this section are crucial for justifying a modular treatment of partial and total correctness in the probabilistic context.

4. Partial and total correctness

In this section we focus explicitly on partial and total correctness; we give our promised formulations of *wp* and *wlp*, both of which are specialisations of the more general *ewp* of the last section, and generalisations of the standard logics [3]. For the new logics we restrict to $\mathscr{P}S$, however: essentially we seek to generalise the discrete domain $\{0,1\}$ (on which predicates are based) to the continuous domain $[0,1]$, so that the transformers give partial rather than absolute judgements of program behaviour. The distingished elements 0 and 1 remain, respectively, as the least and greatest elements under $\Rrightarrow$; those roles will assume significance when we look for least and greatest fixed points.

For a total correctness logic we merely restrict *ewp* to $\mathscr{P}S$ directly, and use the order $\Rrightarrow$.

Definition 4.1. Let r be a program in $\mathscr{M}S$; then the *greatest* preexpectation of program r with respect to postexpectation π in $\mathscr{P}S$, associating 0 with nontermination, is defined

$$wp.r.\pi := ewp.r.\pi.$$

Well definedness follows easily from sublinearity: if π is in $\mathscr{P}S$ then

$$\underline{0} \Rrightarrow ewp.r.\pi \Rrightarrow \underline{1},$$

[9] A more alluring healthiness condition would be that $t.\alpha$ is determined by its positive part $(\alpha \sqcup \underline{0})$ and its negative part $(\alpha \sqcap \underline{0})$; but

$$t.\alpha = t.(\alpha \sqcup \underline{0}) + t.(\alpha \sqcap \underline{0}),$$

does not hold for general probabilistic programs, although it does in the restricted set of standard programs and $\{0,1,-1\}$-valued expectations [18].

so that $wp.r.\pi$ is in $\mathscr{P}S$ also. Moreover Lemma 3.3 shows that this *wp* semantics of programs corresponds to a relational model with the Smyth ordering [20] – nontermination is the worst outcome in both semantics.

For partial correctness we define a probabilistic *wlp*; again we restrict to the subspace $(\mathscr{P}S, \Rrightarrow)$.

Definition 4.2. [10] Let r be a program in $\mathscr{M}S$; then the greatest *liberal* preexpectation of program r with respect to the postexpectation π in $\mathscr{P}S$, associating 1 with nontermination, is

$$wlp.r.\pi := \underline{1} + ewp.r.(\pi - \underline{1}).$$

Again it follows easily from sublinearity of *ewp.r* that for v in $\mathscr{N}S$,

$$\underline{-1} \Rrightarrow ewp.r.v \Rrightarrow \underline{0}$$

and thus since $\pi - \underline{1}$ lies in $\mathscr{N}S$, so does $ewp.r.(\pi - \underline{1})$ from which we deduce that *wlp.r* is a well-defined expectation transformer in $\mathscr{P}S \to \mathscr{P}S$. Also Lemma 3.3 implies that the *wlp* semantics corresponds to a relational model with the Hoare ordering – accordingly nontermination is the best outcome.

Next, we set out alternative (but equivalent) semantics for a simple programming language in Figs. 1 and 2 from which the *wlp* and *wp* semantics can also be derived. Observe that nondeterministic choice $\sqcap$ selects the pointwise minimum between expectations, reflecting the demon's striving for the worst result, whereas probabilistic choice ${}_p\oplus$ selects the weighted average between the two results. In both semantics recursion is dealt with by least fixed points in the appropriate orders: Theorem 3.7 showed that the two orders correspond.

We contrast the two semantics with a small example. Let S be some finite portion of $\mathbb{N}$, and for natural number N write $s := N$ for the assignment taking every initial state to the final state N. The program

$$(s := 0 \;{}_p\oplus s := 1) \;{}_q\oplus \mathbf{abort}$$

illustrates the difference between *wp* and *wlp*. Writing $[s = N]$ for the expectation that evaluates to 1 when s is N and to 0 otherwise, we have

$$wp.((s := 0 \;{}_p\oplus s := 1) \;{}_q\oplus \mathbf{abort}).[s = 0]$$

$$\equiv wp.(s := 0).[s = 0] \;{}_p\oplus\; wp.(s := 1).[s = 0]$$

[10] Via Definition 3.2 we can readily show Definition 4.2 to be identical to

$$wlp.r.\pi.s := \sqcap \left\{ 1 - \int_F (\underline{1} - \pi) \mid F : r.s \right\}, \tag{6}$$

which is a demonic generalisation of the probabilistic *wlp* defined only for *nondemonic* programs by Jones [11]. Morgan [18] shows that (6) also generalises standard *wlp* [3].

$$
\begin{array}{ll}
\textbf{abort}.s & := \{\bar{\perp}\} \\
\textbf{skip}.s & := \{\bar{s}\} \\
(\textbf{assign}\ f).s & := \{\overline{f.s}\} \text{ for function } f \text{ in } S \to S \\
(r\ {}_p\!\oplus r').s & := r.s\ {}_p\!\oplus r'.s \\
(r \sqcap r').s & := \cup_{p:[0,1]} r.s\ {}_p\!\oplus r'.s \\
(r; r').s & := \left\{ \int_F f' \mid F\colon r.s;\ f'\colon S \to \bar{S};\ r' \ni f' \right\} \\
(r\ \textbf{if}\ B\ \textbf{else}\ r').s & := r.s \text{ if } B.s \text{ else } r'.s \\
\textbf{rec}\ \mathscr{B} & := \mu\mathscr{B} \text{ for } \sqsubseteq_{EM}\text{-monotonic } \mathscr{B} \text{ in } \mathscr{M}S \to \mathscr{M}S
\end{array}
$$

For s in S we use $\bar{s}$ to denote the special 'point mass' distribution: applied to a set, $\bar{s}$ returns either one or zero according to s's membership or not.
For ${}_p\!\oplus, \sqcap$ and sequential composition, the Egli–Milner closure should be taken of the right-hand side.

Fig. 1. Probabilistic relational semantics.

$$
\begin{array}{ll}
ewp.\textbf{abort}.\sigma & := \underline{0} \\
ewp.\textbf{skip}.\sigma & := \sigma \\
ewp.(\textbf{assign}\ f).\sigma.s & := \sigma.(f.s) \text{ for function } f \text{ in } S \to S \\
ewp.(r\ {}_p\!\oplus r').\sigma & := (r.\sigma)\ {}_p\!\oplus (r'.\sigma) \\
ewp.(r \sqcap r').\sigma & := ewp.r.\sigma \sqcap ewp.r'.\sigma \\
ewp.(r; r').\sigma & := ewp.r.(ewp.r'.\sigma) \\
ewp.(r\ \textbf{if}\ B\ \textbf{else}\ r').\sigma.s & := ewp.r.\sigma.s \text{ if } B.s \text{ else } ewp.r'.\sigma.s \\
ewp.(\textbf{rec}\ \mathscr{B}) & := \mu\mathscr{F} \text{ where } \mathscr{F} \text{ is the } \sqsubseteq_{\mathscr{F}}\text{-monotonic} \\
 & \quad \text{function such that } \mathscr{F}.(ewp.r) = ewp.(\mathscr{B}.r)
\end{array}
$$

Fig. 2. Probabilistic *ewp* semantics, where σ is in $\mathscr{P}S \cup \mathscr{N}S$ and s is in S.

$$
\begin{aligned}
& {}_q\!\oplus\ wp.\textbf{abort}.[s = 0] \\
\equiv\ & (\underline{1}\ {}_p\!\oplus \underline{0})\ {}_q\!\oplus \underline{0} \quad [s = 0].0 = 1, \textit{etc.} \\
\equiv\ & \underline{pq},
\end{aligned}
$$

indicating that the greatest expectation of termination in state 0 is pq, for all initial states.

The greatest expectation of either termination at 0 or nontermination is found with *wlp*; we have

$$
wlp.((s := 0\ {}_p\!\oplus s := 1)\ {}_q\!\oplus \textbf{abort}).[s = 0]
$$

$$
\begin{aligned}
wp.\mathbf{abort}.\pi &:= \underline{0}\\
wlp.\mathbf{abort}.\pi &:= \underline{1}\\
wp.(\mathbf{rec}\ \mathscr{B}) &:= \mu\mathscr{F} \text{ where } \mathscr{F} \text{ is the } \sqsubseteq_{\mathscr{F}}\text{-monotonic}\\
&\qquad \text{function such that } \mathscr{F}.(ewp.r) = ewp.(\mathscr{B}.r)\\
wlp.(\mathbf{rec}\ \mathscr{B}) &:= \nu\mathscr{F} \text{ where } \mathscr{F} \text{ is the } \sqsubseteq_{\mathscr{F}}\text{-monotonic}\\
&\qquad \text{function such that } \mathscr{F}.(ewp.r) = ewp.(\mathscr{B}.r)
\end{aligned}
$$

Fig. 3. Probabilistic *wp* and *wlp* semantics, where π is in $\mathscr{P}S$ and s is in S. The least μ or greatest fixed point ν is used for recursion.

$$
\begin{aligned}
&\equiv (\underline{1}\ {}_p{\oplus}\underline{0})\ {}_q{\oplus}\underline{1}\\
&\equiv \underline{pq + 1 - q}.
\end{aligned}
$$

Thus the *wp* observation gives the greatest guaranteed probability of termination at 0 – and nontermination guarantees nothing. The *wlp* observation, on the other hand, returns the probability that either 0 is reached or the program fails to terminate – the usual interpretation for partial correctness.

Perhaps the most telling difference between *wp* and *wlp* lies in the analysis of an explicit recursion. It is easy to show the *wp* semantics of a looping program is given by the least fixed point of a monotonic function in the $\Rrightarrow$ order lifted to transformers, whereas in the *wlp* semantics it is the greatest fixed point. This follows from Definition 4.2 since the least fixed point of a $\sqsubseteq_T$-monotonic function becomes specialised first to $\Lleftarrow$ on $\mathscr{N}S \to \mathscr{N}S$, and finally is shifted to $\mathscr{P}S \to \mathscr{P}S$ by applying "$\underline{1}+$".

It is easily checked that specialising Fig. 2 to *wlp* and *wp* produces the only the changes shown in Fig. 3.

5. Invariant/variant reasoning for loops

In this section we use the *wp* and *wlp* logics to generalise a rule allowing the separation of invariant/variant reasoning for probabilistic programs. It forms the second main contribution of this paper. We begin with a discussion of the standard case.

The standard rule follows from the so-called 'coupling law' [10]:

$$wlp.\mathrm{prog}.A \wedge wp.\mathrm{prog}.true \Rightarrow wp.\mathrm{prog}.A, \tag{7}$$

where *prog* is a program and A a predicate and we are using (though only here) the original meanings for *wp* and *wlp* [3], with $\Rightarrow$ for 'implies at all states'.

Law (7) implies that *wp.prog* and *wlp.prog* agree on initial states from which termination is guaranteed, and thus it underlies the practical treatment of looping programs – to prove total correctness of an iteration the work is divided between ensuring partial correctness (with a loop invariant), and an independent termination argument (with a variant). It is the probabilistic coupling Theorem 5.2 that allows a similar treatment for probabilistic looping programs. Crucial here is the idea that probabilistic judgements may be modularised, even for recursion where the final distribution may be made up of many small probabilistic choices resolved in preceeding recursive steps. In standard semantics modular reasoning is possible because *wp* and *wlp* are conjunctive (provided by Dijkstra's healthiness condition). For probabilistic semantics, our characterisation Theorem 3.11 supplies above all sublinearity, and we consider next how to use it to replace conjunction of predicates with an appropriate alternative defined for expectations.

We define *probabilistic conjunction* [25] for nonnegative expectations π, π':

$$\pi \,\&\, \pi' := (\pi + \pi' - \underline{1}) \sqcup \underline{0}, \tag{8}$$

where $\sqcup$ is pointwise maximum between expectations. Probabilistic conjunction reduces to ordinary conjunction when specialised to predicates.[11] Its importance in probabilistic reasoning is that it subdistributes through both *wp* and *wlp* images of programs – another consequence of sublinearity.[12]

Lemma 5.1. *For r in $\mathscr{M}S$ and π, π' in $\mathscr{P}S$,*

$$wp.r.(\pi \,\&\, \pi') \Leftarrow wp.r.\pi \,\&\, wp.r.\pi'$$

$$wlp.r.(\pi \,\&\, \pi') \Leftarrow wlp.r.\pi \,\&\, wlp.r.\pi'.$$

Proof. Sublinearity (with $a = b = c = 1$) and monotonicity of *ewp.r*, and Definitions 4.1, 4.2. □

Next, we deal with coupling – Theorem 5.2, generalising (7), is the main result of this section.

Theorem 5.2. *For r in $\mathscr{M}S$ and π, π' in $\mathscr{P}S$,*

$$wp.r.\pi \,\&\, wlp.r.\pi' \Rrightarrow wp.r.(\pi \,\&\, \pi').$$

[11] If π, π' in $\mathscr{P}S$ take only the extreme values 0 and 1, then

$$(\pi \,\&\, \pi').s = \begin{cases} 1 & \text{if } \pi.s = 1 \text{ and } \pi'.s = 1, \\ 0 & \text{otherwise.} \end{cases}$$

[12] One might have guessed that $\sqcap$ is the appropriate generalisation of $\wedge$ – but $\sqcap$ does not (even sub-) distribute [11, 25].

Proof.

$$
\begin{aligned}
& wp.r.(\pi \,\&\, \pi') \\
\equiv\ & ewp.r.(\pi \,\&\, \pi') \quad \text{Definition 4.1} \\
\equiv\ & ewp.r.((\pi + \pi' - \underline{1}) \sqcup \underline{0}) \quad (8) \\
\Leftarrow\ & ewp.r.(\pi + \pi' - \underline{1}) \quad \text{monotonicity} \\
\Leftarrow\ & ewp.r.\pi + ewp.r.(\pi' - \underline{1}) \quad \text{sublinearity} \\
\equiv\ & ewp.r.\pi + \underline{1} + ewp.r.(\pi' - \underline{1}) - \underline{1} \quad \text{arithmetic} \\
\equiv\ & wp.r.\pi + wlp.r.\pi' - \underline{1} \quad \text{Definition 4.2; Definition 4.1.}
\end{aligned}
$$

Having established $wp.r.(\pi \,\&\, \pi') \Leftarrow wp.r.\pi + wlp.r.\pi' - \underline{1}$, we conclude by taking $\sqcup\underline{0}$ on both sides: since on the left it has no effect, we achieve our result. □

As a corollary we recover the standard rule (generalised) for combining partial and total correctness.

Corollary 5.3. *For r in $\mathscr{M}S$ and π in $\mathscr{P}S$,*

$$wlp.r.\pi \,\&\, wp.r.\underline{1} \Rrightarrow wp.r.\pi.$$

Proof. Theorem 5.2 and that $\pi \equiv \pi \,\&\, \underline{1}$ for π in $\mathscr{P}S$. □

As a special case note that the *wlp* result implies the *wp* result at those states from which termination occurs with probability 1 – where $wp.r.\underline{1} \equiv \underline{1}$ – because $(\&\underline{1})$ is the identity.

Our results so far have been specialised to discrete distributions over a countable state space S, using the flat domain. In the next section we show that with only a little more work, and provided our underlying state space has a more sophisticated structure (than that of the flat domain), the constructions apply even to continuous distributions, over the reals.

6. Continuous probability distributions

In this section we indicate how our results can be extended to continuous probability distributions. The goal is to obtain a program semantics supporting probabilistic assignments in which the selection ranges over probability distributions, such as the uniform distribution over a compact subset of the reals, together with a quantitative logic in which the expressions are integrable real-to-real functions. Kozen [13] explores the relation between the Smyth order on (deterministic) programs and continuous probability distributions via a metric on measures – the measures themselves are (as usual) based on the Borel field generated by 'open' intervals of the real line. For the present

more general context (Egli–Milner order and demonically nondeterministic programs) we must make finer distinctions between programs and in consequence we use instead a domain of 'continuous valuations'. [13]

In order to apply the decomposition results of Section 2, we need to find a complete partial order of measures (over the reals) satisfying the assumptions of Corollary 2.2. Because of that, the traditional approach to measure theory is not appropriate, as it fails to be ω-continuous. [14] Fortunately, however, the more general presentation of Edalat [4] (in which we find standard measure theory as a special case) does not have this problem, and thus we shall use that for the construction of our program semantics. We begin by reviewing the main ingredients of his work.

Edalat's idea is to consider the Borel field generated by (open) sets of *compact intervals* (rather than of points).

Definition 6.1. We define the directed-complete partial order of *compact intervals* $(\mathscr{V}, \sqsubseteq)$ to be

$$\mathscr{V} := \{[a,b] \mid a,b\colon \mathbb{R};\ a \leqslant b\},$$

where $[a,b]$ denotes the closed interval of the real line between a and b, and for intervals $[a,b]$ and $[a',b']$ we define

$$[a,b] \sqsubseteq [a',b'] \quad \text{iff} \quad [a,b] \supseteq [a',b'].$$

The Scott topology on $(\mathscr{V}, \sqsubseteq)$ is given by the basic open sets of the form

$$\{[a,b] \mid a,b\colon \mathbb{R};\ [a,b] \subseteq (x,y)\},$$

for any $x < y$ where (x,y) denotes the open interval between x and y. We shall denote this Scott topology by $\Omega\mathscr{V}$.

Next, we define the set of 'continuous valuations' over $\mathscr{V}$ – they will be used to model our continuous probability measures over $\mathbb{R}$. A function $G : \Omega\mathscr{V} \to \mathbb{R}$ is said to be a (*continuous*) *valuation* over $\mathscr{V}$ if it satisfies the following properties [8]:

$$G.(X \cup Y) = G.X + G.Y - G.(X \cap Y) \quad \text{modularity}$$

$$G.\emptyset = 0 \quad \text{strictness}$$

$$G.(\cup_{i:\mathscr{I}} X_i) = \sqcup_{i:\mathscr{I}} G.X_i \quad \text{continuity},$$

where $\mathscr{I}$ is a $\subseteq$-directed set of open sets and X, Y are open sets.

Roughly speaking, valuations differ from measures in that they are defined only on the open sets of the topology; but they can be extended to measures on the full

[13] In this section we shall use the term 'continuous' in several slightly different contexts, although all relate to the general topological notion.

[14] To see that it fails, we consider the uncountable set of point distributions, and observe that they are pairwise incomparable and that each must be approximated by a unique element in any (hence uncountable) basis.

Borel field (generated by the topology). Valuations also have an order induced by the underlying $\sqsubseteq$ on $\mathscr{V}$: for valuations G and G' we say that

$$G \sqsubseteq G' \quad \text{iff} \quad (\forall X : \Omega\mathscr{V})(G.X \leqslant G'.X).$$

(Note that this definition reduces to Definition 2.3 in the context of Section 2, since in the Scott topology of the flat domain all sets are open, and in particular singleton sets are.) We denote the space of continuous valuations over $\mathscr{V}$ by $(\bar{\mathscr{V}}, \sqsubseteq)$.

An important property of continuous valuations is that they can be approximated by directed sets of 'simple valuations' [4]. A valuation F is said to be *simple* if it can be expressed as a finite sum, that is

$$F = a_1\overline{d_1} + \cdots + a_n\overline{d_n},$$

where $a_1, \ldots, a_n$ are nonnegative reals whose sum is no more than 1, and $\bar{d}$ denotes a point measure, defined

$$\bar{d}.X := \begin{cases} 1 \text{ if } d \in X \\ 0 \text{ otherwise} \end{cases}$$

for any subset X of $\mathscr{V}$.

Denoting by $\mathscr{E}\mathscr{V}$ the continuous functions[15] $\mathscr{V} \to \mathbb{R}$, we define integration of a function β in $\mathscr{E}\mathscr{V}$ with respect to a simple valuation (as for F above):

$$\int_F \beta := \sum_{1 \leqslant i \leqslant n} a_i(\beta.d_i).$$

Next, for any G, we define the integral

$$\int_G \beta := \sqcup \left\{ \int_{F'} \beta \mid G' : \mathscr{G} \right\},$$

where $\mathscr{G}$ is any directed set of simple valuations with limit G. Well definedness is proved elsewhere [4, 5].

Our interest in valuations is twofold:

- Ordinary Lebesgue integration (and Riemann–Stieltjes integration, the technique for evaluating expectations with respect to probability distributions) can be formulated in terms of valuations and continuous functions [4]: for example an (ordinary) continuous function[16] $\beta : \mathbb{R} \to \mathbb{R}$ extends to an (upper) continuous function $\hat{\beta} \in \mathscr{E}\mathscr{V}$, by

$$\hat{\beta}.[a, b] := \sqcap\{\beta.x \mid x : [a, b]\};$$

[15] *Upper continuous* means that $\beta^{-1}.(a, \infty) \in \Omega\mathscr{V}$ for all a in $\mathbb{R}$ and *lower continuous* means that $\beta^{-1}.(-\infty, a) \in \Omega\mathscr{V}$ for all a in $\mathbb{R}$. We use *continuous* here to mean either upper or lower continuous (or both).

[16] We use 'ordinary continuous function' for real-to-real functions that are continuous in the usual sense of real analysis.

likewise, an ordinary (probability) measure is found more generally as the extension of a directed limit of simple valuations. The uniform distribution over the compact set $[a,b]$ for example is the limit (as n approaches infinity) of the chain of simple valuations,

$$\overline{[a,b]} \sqsubseteq 1/2\overline{[a,(b-a)/2]} + 1/2\overline{[(b-a)/2,b]} \sqsubseteq \cdots$$
$$\sum_{0 \leqslant k \leqslant n-1} 1/n\overline{[a+(b-a)k/n, a+(b-a)(k+1)/n]} \sqsubseteq \cdots$$

- $(\bar{\mathscr{V}}, \sqsubseteq)$ is an ω-continuous directed-complete partial order: we take as the basis the subset of the simple valuations which only involve rationals (the point valuations are given by intervals with rational endpoints, and the coefficients in the finite sum are also rational).

Thus $(\bar{\mathscr{V}}, \sqsubseteq)$ enjoys ω-continuity whilst supporting standard integration of probability measures – and those two features together make it appropriate for our application to program semantics, for which we now set out the details.

Definition 6.2. The space of programs $(\mathscr{MV}, \sqsubseteq_{EM})$ consists of the set $\mathscr{MV}$ of *continuous functions* $\mathscr{V} \to \mathscr{C}\bar{\mathscr{V}}$, and the order between programs $r, r' : \mathscr{MV}$ is defined

$$r \sqsubseteq_{EM} r' \quad \text{iff} \quad (\forall d : \mathscr{V})(r.d \sqsubseteq_{EM} r'.d).$$

Recall that $\mathscr{C}\bar{\mathscr{V}}$ is the space of lenses; the results of Section 2 now apply. Notice also that we restrict to continuous functions, insisting that

$$r.(\sqcup \mathscr{D}) = \sqcup_{EM} \{r.d \cdot d : \mathscr{D}\}$$

for a directed set $\mathscr{D}$ in $\mathscr{V}$, and where the limit on the left-hand side is taken in $\mathscr{V}$ and in $\mathscr{MV}$ on the right. (In the flat domain and thus in Definition 3.1 that condition is redundant.)

The program logic is similar to Definition 3.2.[17]

Definition 6.3. Let r be a program in $\mathscr{MV}$ taking initial intervals in $\mathscr{V}$ to sets of final valuations over $\mathscr{V}$. Then the *greatest* preexpectation at interval d of program r, with respect to postexpectation β in $\mathscr{EV}$, is defined

$$ewp.r.\beta.d := \sqcap \left\{ \int_F \beta \mid F : r.s \right\}.$$

Likewise the order on transformers is as for Definition 3.4:

$$t \sqsubset_{EM} t' \quad \text{iff} \quad (\forall \pi : \mathscr{PV})(t.\pi \Rrightarrow t'.\pi) \wedge (\forall \pi : \mathscr{NV})(t.\pi \Lleftarrow t'.\pi),$$

where $\mathscr{PV}$ and $\mathscr{NV}$ are, respectively, the nonnegative and nonpositive subsets of $\mathscr{EV}$.

[17] For simplicity, we consider a program with one variable taking values over $\mathscr{V}$.

$$
\begin{array}{lll}
ewp.\mathbf{abort}.\sigma & := & \underline{0} \\
ewp.\mathbf{skip}.\sigma & := & \sigma \\
ewp.(\mathbf{assign}\ f).\sigma.d & := & \sigma.(f.d) \text{ for continuous function } f \text{ in } \mathbb{R} \to \mathbb{R} \\
ewp.(\mathbf{assign}\ \mu).\sigma & := & \int_{\mu} \sigma \text{ for continuous valuation } \mu \text{ in } \bar{\mathscr{V}} \\
ewp.(r \sqcap r').\sigma & := & ewp.r.\sigma \sqcap ewp.r'.\sigma \\
ewp.(r; r').\sigma & := & ewp.r.(ewp.r'.\sigma) \\
ewp.(r\ \mathbf{if}\ B\ \mathbf{else}\ r').\sigma.d & := & ewp.r.\sigma.d \text{ if } d \subseteq B^o \\
 & & ewp.r.\sigma.d \text{ if } d \subseteq (\mathbb{R} - B)^o \\
 & & \underline{0}, \text{otherwise} \\
ewp.(\mathbf{rec}\ \mathscr{B}) & := & \mu\mathscr{F} \text{ where } \mathscr{F} \text{ is the } \sqcap_\tau\text{-monotonic} \\
 & & \text{function such that } \mathscr{F}.(ewp.r) = ewp.(\mathscr{B}.r)
\end{array}
$$

Fig. 4. Probabilistic *ewp* semantics for real state spaces, where σ is in $\mathscr{PV} \cup \mathscr{NV}$ and d is in $\mathscr{V}$. Also $B \subseteq \mathbb{R}$ is an interval and X^o denotes the largest open set contained in X.

With these definitions we have the analogues of Sections 3 and 4:

- For program r in $\mathscr{MV}$, the transformer $ewp.r$, is well defined as a function $\mathscr{EV} \to \mathscr{EV}$. It is directed-continuous and sublinear.
- *ewp* is an embedding of programs into transformers:

$$r \sqsubset_{EM} r' \quad \text{iff} \quad ewp.r \sqsubset_{EM} ewp.r'.$$

- $wp.r.(\pi \& \pi') \Rrightarrow wp.r.\pi \& wlp.r.\pi$ holds for $r \in \mathscr{MV}$ and upper continuous $\pi, \pi' \in \mathscr{PV}$. (The proof of Theorem 5.2 goes through once we note that if π, π' are (upper) continuous, then so are $\pi \& \pi'$ and $\pi + \pi' - \underline{1}$.)

We end this section by defining the transformer semantics for programs over the reals, set out in Fig. 4.

The most notable difference between Figs. 4 and 2 is the semantics for conditional choice where, for continuous distributions, there is the possibility of aborting if the 'input interval' cannot be determined to lie entirely within or without the condition B. That choice is explained mathematically by the need to define a continuous transformer, and operationally by the intuition that programs cannot in general compute the function 'equals' over the reals. Practically, it means that the only semantics we can give to the program

$$(x := 1)\ \mathbf{if}\ x \geqslant 0\ \mathbf{else}\ (x := 0),$$

implies that it must abort at $x = 0$ when x ranges over the type of reals. Escardo [6] discusses this and other issues arising in semantics of real-valued computations.

7. Conclusion

Our main theoretical contribution is a logical characterisation of an operational model based on the Plotkin powerdomain of probability distributions; it is applicable to probabilistic sequential programs and can be used in the analysis of probabilistic distributed algorithms. The logic extends Morgan [20], Jones [11] and Kozen [14], for among the concepts probability, nondeterminism and nontermination, it encapsulates all three.

The healthiness conditions embodying the characterisation guide our generalisation of the rule modularising partial and total correctness, and as such represents a considerable step in the available proof methods for probabilistic algorithms; that rule forms our second contribution.

Jones [11] defines a partial correctness logic based on expectations, but only for nondemonic programs, and she does not discuss the healthiness conditions on which the applicability of such logics (as calculational tools) depends. It was the realisation [20] that adding nondeterminism to Kozen's model corresponds to a weakening of the additive property of his logic to sublinearity that makes proofs such as in Theorem 5.2 and those in [17] reduce to simple arithmetic arguments. The use of general expectations (thus superseding purely nonnegative expectations [20]) leads to an even simpler presentation of sublinearity – the more useful of the three healthiness conditions described here.

A key feature of the operational model is the imposition of convexity (linear interpolation) between distributions, for it allows programs to be characterised exactly at the logical level (Theorem 3.7). That feature is not usually present in other treatments of probability and nondeterminism: Segala [24] for example has no *wp*/*wlp*-style program logic in his model for distributed systems (though he does consider temporal logic), and indeed he does not explore the relationship between logical models and partial orders as we do here.

However, convexity has more uses than logical characterisation: the lack of the convex healthiness condition, for example, means that even at the operational level, the reverse subset inclusion relation between sets of outputs (the usual definition for program refinement) must be augmented by the explicit addition of the law $\sqcap \sqsubseteq {}_p\!\oplus$ [24]. With convexity the law is automatic. More generally, operational models that enforce convexity also satisfy other nice algebraic laws between program constructs [9].

There are two immediate applications. The first is the discovery of proof rules for loops in which partial and total correctness are separated: with *wp* and *wlp* together, using the theory of this paper, it can be shown [17] that

$$I \sqcap G \Rrightarrow wp.body.I \quad \text{probabilistic invariant preserved by loop body}$$

is sufficient for $I \Rrightarrow wp.(\mathbf{do}\ G \to body\ \mathbf{od}).(I \sqcap \neg G)$ provided $I \Rrightarrow T$, where T gives for each initial state the probability of the loop's termination. That rule is standard for nonprobabilistic programs and thus we give for the first time its generalisation.

The second application concerns abstraction. In some cases it is useful to analyse a probabilistic algorithm by 'converting' all its probabilistic choices to demonic, and

showing it remains correct even then – but only if it terminates. That requires *wlp*. A separate (explicitly probabilistic) argument shows the chance of termination. Putting the correct-if-terminates (*wlp*) and the termination (*wp*) results together uses the techniques we have presented here. Algorithms falling into that category typically use randomisation as a method for searching a large space of potential witnesses, and examples include finding perfect hash functions and finding irreducible polynomials [26].

Appendix A. Separation lemmas

Lemma A.1. The separating hyperplane lemma. *Let C be a convex and (limit-) closed subset of $\mathbb{R}^N$, and let F be a point in $\mathbb{R}^N$ that does not lie in C. Then there is a separating hyperplane α with F on one side of it and all of C on the other.*

Proof. See for example Trustrum [27]. □

Our use in Lemma 3.6, for example, is based on interpreting the 'projection' (see below) of a valuation F as a point in $\mathbb{R}^N$, and an expectation α as the collective normal of a family of parallel hyperplanes. The integral $\int_F \alpha$ then gives the constant term for the α-hyperplane that passes through the projection of the point F, where in this context

$$\int_F \alpha := \sum_{1 \leqslant i \leqslant N} \alpha.i \times F.i.$$

Along the same lines, for a convex region K in $\mathbb{R}^N$, the minimum

$$\sqcup \left\{ \int_F \alpha \mid F : K \right\}$$

can be interpreted as the constant term for the α-hyperplane that touches K, with its normal pointing into K.

Thus, when specialised for the applications in this paper, the lemma implies a more general separation: that if $F \notin C$ for some compact and up-closed (or Scott-closed) convex set of valuations C, then there is an expectation α with

$$\int_F \alpha < \sqcap \left\{ \int_{F'} \alpha \mid F' : C \right\}$$

with the range of α above specialising to $\mathscr{P}D$ (or $\mathscr{N}D$) and the integral $\int_F \alpha$ is defined more generally in the context of D. In the following lemmas, we set out the details applicable to each of S and $\mathscr{V}$.

Lemma A.2. *Given a Scott compact, up-closed set C of continuous valuations (or distributions), over an ω-continuous domain D, and a valuation F that does not lie*

in C, then there is a continuous function $\alpha : D \to \mathbb{R}_{\geqslant}$ such that

$$\int_F \alpha < \sqcap \left\{ \int_G \alpha \mid G : C \right\}.$$

Proof. Let ΩD be the set of Scott-open sets, then we have from the definition $\sqsubseteq$ on valuations (Definition 2.3 or Definition 6.1)

$$\downarrow \{F\} = \bigcap_{X:\Omega D} \{G \mid G : \bar{D}; F.X \geqslant G.X\},$$

an intersection of Scott-closed sets (it is both down-closed and directed-complete).

Thus we have,

$$\begin{aligned} &\subseteq \bigcup_{X:\Omega D}^{G} \{G \mid G : \bar{D}; F.X < G.X\} \quad C \text{ up-closed, } F \notin C \\ &= \bigcup_{X:\Omega D, \varepsilon > 0} \{G \mid G : \bar{D}; F.X + \varepsilon < G.X\} \quad \text{property of the reals.} \end{aligned}$$

The last step gives a Scott-open cover of C, and now compactess gives us a finite sub-cover:

$$C \subseteq O_1 \cup \cdots \cup O_n,$$

where we have defined

$$O_i := \{G : \bar{D} \mid F.X + \varepsilon < G.X_i\}.$$

(We do not need to vary ε, since decreasing it increases the open set, thus the finite subcover given by compactess can be replaced by another in which the value of $\varepsilon > 0$ is the same in all sets, as shown.)

Consider now the element and subsets in $\mathbb{R}^n$ defined:

$$F' := (F.X_1, \ldots, F.X_n)$$

$$C' := \{(G.X_1, \ldots, G.X_n) \mid G : C\}$$

$$O_i' := \{(G.X_1, \ldots, G.X_n) \mid G : X_i\}.$$

By choice of O_i we have that $F' \notin O_i'$ for any i and thus (since $\varepsilon > 0$) that $F' \notin sc.C'$, where $sc.C'$ denotes the closure of C' in the Euclidean metric on $\mathbb{R}^n$; moreover $sc.C'$ is convex and up-closed (because C is). Now applying Lemma A.1 to $sc.C'$ we find a (nonnegative) hyperplane $(b_1, \ldots, b_n)$ such that

$$\sum_{1 \leqslant i \leqslant n} b_i \times F.X_i < \sqcap \left\{ \sum_{1 \leqslant i \leqslant n} b_i \times G.X_i \mid G : C \right\}. \tag{9}$$

Finally, we define

$$\beta := b_1 \chi_{X_1} + \cdots + b_n \chi_{X_n},$$

where χ_X is the characteristic function on X which evaluates to 1 at points in X and zero elsewhere. It is a continuous function if X is Scott-open, and (nonnegative) sums of such functions are also continuous.

The lemma now follows since $\int_G \beta = \sum_{1 \leqslant i \leqslant n} b_i \times GX_i$ for any valuation G, which on substitution for the summations in (9) gives us the required inequality. □

The alternative case, that of separating a point from a Scott-closed subset, can be proved similarly, and the details appear elsewhere [16].

Lemma A.3. *Let D be one of S or $\mathscr{V}$. Given a Scott-closed (hence down closed) set C of continuous valuations (or distributions), and a valuation F that does not lie in C, then there is an integrable function $\beta : D \to \mathbb{R}_{\geqslant}$ such that*

$$\int_F \beta > \sqcup \left\{ \int_G \beta \mid G{:}\ C \right\}.$$

Appendix B. Facts and definitions from domain theory

We summarise here Abramsky and Jung's presentation [1] of facts from domain theory, giving page numbers where appropriate. Assume $(D, \sqsubseteq)$ is a complete partial order – i.e. that every directed (defined below) set has a least upper bound:

1. *Up-, down-closure*: For subset A of D we define its *up-closure* $\uparrow A$ to be the set

   $$\{d : D \mid (\exists a{:}A)(a \sqsubseteq d)\}.$$

   Similarly, we define its *down-closure* $\downarrow A$ to be the set

   $$\{d{:}\ D \mid (\exists a{:}\ A)(a \sqsupseteq d)\}.$$

2. *Egli–Milner closure*: For a subset A of D we define its *Egli–Milner closure* as $\uparrow A \cap \downarrow A$.
3. *Smyth order* (p. 97): The *Smyth order* $\sqsubseteq_S$ on $\mathbb{P}D$, for subsets A, A' of D, is given by

   $$A \sqsubseteq_S A' \quad \text{iff} \quad A' \subseteq \uparrow A.$$

4. *Hoare order* (p. 97): The *Hoare order* $\sqsubseteq_H$ on $\mathbb{P}D$, for subsets A, A' of D, is given by

   $$A \sqsubseteq_H A' \quad \text{iff} \quad A \subseteq \downarrow A'.$$

5. *Egli–Milner order*: The Egli–Milner order $\sqsubseteq_{EM}$ between subsets combines the Smyth (Definition B3) and Hoare (Definition B4) order. For subsets A, A' of D we define

   $$A \sqsubseteq_{EM} A' \quad \text{iff} \quad \uparrow A \supseteq A' \text{ and } A \subseteq \downarrow A'.$$

6. *Up-directed* (Definition 2.1.8, p. 10): A subset A of D is *up-directed* (or simply *directed*) iff for any u,v in A there is a w also in A such that $u \sqsubseteq w$ and $v \sqsubseteq w$. We write $\sqcup A$ for the least upper bound of A (if it exists).
7. *Down-directed*: A subset A of D is *down-directed* iff for any u,v in A there is a w also in A such that $w \sqsubseteq u$ and $w \sqsubseteq v$. We write $\sqcap A$ for the greatest lower bound of A (if it exists).
8. *Way-below* (Definition 2.2.1, p. 15): The *way-below* relation $\ll$ on D is defined as follows: for u,v in D we say $u \ll v$ iff for all up-directed subsets A of D with $v \sqsubseteq \sqcup A$ there is some w in A with $u \sqsubseteq w$. We also say u *approximates* v iff $u \ll v$.
9. *Basis* (Definition 2.2.3, p. 16): A *basis* for D is a subset B such that every element of D is the $\sqcup$-limit of the elements way below it in B.
10. *ω-Continuity* (Definition 2.2.6, p. 17): D is *ω-continuous* if it has a countable basis.
11. *Scott topology* (Definition 2.3.1, p. 28): A subset A of D is *Scott-closed* if it is down-closed and clused under suprema od directed limits. Complements of Scott-closed sets are *Scott-open*. We write $sc.A$ for the smallest Scott-closed set containing A.
12. *Scott compact*: A subset A is *Scott compact* if any Scott-open cover of A has a finite sub-cover.
13. *Lens* (Definition 6.2.15, p. 100): We define the lenses of D, $Lens(D)$, to be the set of non-empty sets arising as the intersection of a Scott-closed subset and a Scott-compact, up-closed subset (of D).
14. *Ideal* (p. 10): A subset I is an *ideal* if it is directed and down-closed.
15. *Plotkin powerdomain* (Theorem 6.2.3, p. 95): The *Plotkin powerdomain* of D with basis B is given by the ideal (Definition B14) completion of

$$(\mathscr{F}(B), \ll_{EM}),$$

where $\mathscr{F}(B)$ denotes the set of finite, nonempty subsets of B.
16. *Topological Egli–Milner order* (Definition 6.2.16, p. 101): Define the *topological Egli–Milner order* $\sqsubseteq_{TEM}$ on the set of Egli–Milner closed subsets, $Lens(D)$, of D as follows:

$$X \sqsubseteq_{TEM} Y \quad \text{iff} \quad \uparrow X \supseteq Y \wedge X \subseteq sc.(\downarrow Y).$$

17. *Smyth-*, *Hoare-equivalence*: Two subsets A, A' of D are *Smyth equivalent* if $\uparrow A = \uparrow A'$; they are *Hoare-equivalent* if $\downarrow A = \downarrow A'$.
18. *Consistent complete*: A partially ordered set is said to be *consistently complete* if whenever elements d, d', d'' satisfy $d \leqslant d''$ and $d' \leqslant d''$ then there is a least element e such that $d \leqslant e$ and $d' \leqslant e$.

Theorem B.1 (Theorem 6.2.19, p. 101). *If D is an ω-continuous complete partial order, its Plotkin powerdomain is isomorphic to $(Lens(D), \sqsubseteq_{TEM})$.*

References

[1] S. Abramsky, A. Jung, Domain theory, in: Dov M. Gabbay S. Abramsky, T.S. Maibaum (Eds.), Handbook of Logic and Computer Science, Vol. 3, Oxford, Clarendon Press, 1994, p. 1–168.
[2] K.M. Chandy, J. Misra, Parallel Program Design: A Foundation, Addison-Wesley, Reading, MA, 1988.
[3] E.W. Dijkstra, A Discipline of Programming, Prentice-Hall, Englewood Cliffs, NJ, 1976.
[4] A. Edalat, Domain theory and integration, Theoret. Comput. Sci. 151 (1995) 163–193.
[5] A. Edalat, S. Negri, The generalised Riemann integral on locally compact spaces, Topology Appl. 89 (1998) 121–150.
[6] M.H. Escardo, Pcf extended with real numbers, Theoret. Comput. Sci. 162 (1996) 79–115.
[7] Y.A. Feldman, D. Harel, A probabilistic dynamic logic, J. Comput. System Sci. 28 (1984) 193–215.
[8] G. Grimmett, D. Welsh, Probability: an Introduction, Oxford, Clarendon Press, 1986.
[9] J. He, K. Seidel, A.K. McIver, Probabilistic models for the guarded command language, Sci. Comput. Programming 28 (2,3) (1997) 171–192.
[10] W.H. Hesselink, Programs, Recursion and Unbounded Choice, Number 27 in Cambridge Tracts in Theoretical Computer Science, Cambridge University Press, Cambridge, UK, 1992.
[11] C. Jones, Probabilistic nondeterminism. Monograph ECS-LFCS-90-105, Ph.D. Thesis, Edinburgh University Edinburgh, UK, 1990.
[12] A. Jung, private communication, 1997.
[13] D. Kozen, Semantics of probabilistic programs, J. Comput. System Sci. 22 (1981) 328–350.
[14] D. Kozen, A probabilistic PDL, in: Proc. 15th ACM Symp. on Theory of Computing, ACM, New York, 1983.
[15] A.K. McIver, C.C. Morgan, Demonic, angelic and unbounded choices in probabilistic programs, Tech. Report PRG-TR-5-96, Programming Research Group, March 1996. See PPT athttp [23]; accepted for publication in Acta Informatica.
[16] A.K. McIver, C.C. Morgan, Partial correctness for probabilistic programs. Tech. Report, Programming Research Group, 1996. See PCFPP athttp [23].
[17] C.C. Morgan, Proof rules for probabilistic loops, in: H. Jifeng, J. Cooke, P. Wallis (Eds.), Proc. BCS-FACS Seventh Refinement Workshop, Workshops in Computing, Springer, Berlin, 1996. http://www.springer.co.uk/ewic/workshops/7RW.
[18] C.C. Morgan, A.K. McIver, Unifying *wp* and *wlp*, Inform. Process. Lett. 20(3) (1996) 159–164. Also available via http [23].
[19] C.C. Morgan, A.K. McIver, J.W. Sanders, Probably Hoare? Hoare probably! 1999.
[20] C.C. Morgan, A.K. McIver, K. Seidel, Probabilistic predicate transformers, ACM Trans. Programming Languages Systems 18 (3) (1996) 325–353.
[21] G. Nelson, A generalization of Dijkstra's calculus, ACM Trans. Programming Languages Systems 11 (4) (1989) 517–561.
[22] PSG (Probabilistic Systems Group) Collected Reports. http://www.comlab.ox.ac.uk/oucl/groups/probs/bibliography.html.
[23] J.R. Rao, Building on the UNITY experience: compositionality, fairness and probability in parallelism, Ph.D. Thesis, University of Texas at Austin, Austin, TX, 1992.
[24] R. Segala, Modeling and verification of randomized distributed real-time systems, Ph.D. Thesis, 1995.
[25] K. Seidel, C.C. Morgan, A.K. McIver, An introduction to probabilistic predicate transformers, Tech. Report PRG-TR-6-96, Programming Research Group, February 1996, Also available via http [23].
[26] S.A. Smolka, G. Gupta, S. Bhaskar, On randomization in sequential and distributed systems, ACM Comput. Surveys 26 (1) (1994) 7–86.
[27] K. Trustrum, Linear Programming, Library of Mathematics, Routledge & Kegan Paul, London, 1971.