Acta Informatica 37, 329–354 (2001)

Demonic, angelic and unbounded probabilistic choices in sequential programs

A.K. McIver[⋆], **C. Morgan**[⋆⋆]

Programming Research Group, Oxford University, 8-11 Keble Rd., Oxford OX1 3QD, United Kingdom (e-mail: anabel@comlab.ox.ac.uk, carrollm@unsw.edu.au)

Received: 10 November 1998 / 7 September 2000

Abstract. Probabilistic predicate transformers extend standard predicate transformers by adding probabilistic choice to (transformers for) sequential programs; demonic nondeterminism is retained. For finite state spaces, the basic theory is set out elsewhere [17], together with a presentation of the probabilistic 'healthiness conditions' that generalise the 'positive conjunctivity' of ordinary predicate transformers.

Here we expand the earlier results beyond ordinary conjunctive transformers, investigating the structure of the transformer space more generally: as Back and von Wright [1] did for the standard (non-probabilistic) case, we nest deterministic, demonic and demonic/angelic transformers, showing how each subspace can be constructed from the one before. We show also that the results hold for infinite state spaces.

In the end we thus find characteristic healthiness conditions for the hierarchies of a system in which deterministic, demonic, probabilistic and angelic choices all coexist.

1 Introduction

Probabilistic predicate transformers, or 'expectation transformers' as we call them here, were introduced by Kozen [6] as a logic for imperative probabilistic programming. Recently [17] we extended Kozen's work to include

⋆ McIver is a member of the Programming Research Group at the University of Oxford, and is supported by the EPSRC.

⋆⋆ Morgan was partially supported during this work by the Dutch *Specification and Transformation of Programs* (STOP) project.

demonic nondeterminism, thus placing it on a par with the general treatment of imperative nondeterminism in the 'standard' (non-probabilistic) case [2].

In this paper we make two contributions. First, we investigate systematically the structure of the whole expectation transformer space – not just the conjunctive-like part of it – revealing a hierarchy of program classes (Fig. 9 below), with the more complex being built from the simpler ones. (Back and von Wright [1] reveal a similar hierarchy for standard programs.) Further, we associate with each class of program a characteristic 'healthiness condition' in the expectation logic, showing it to be necessary and sufficient for membership of the class.

Second, we show that both our earlier work [17] and the current results extend to infinite state spaces, provided a continuity assumption is made.

Programs as standard predicate transformers can be presented as nested classes of increasing sophistication: the *deterministic* programs deliver only one final state for any initial state, and are characterised by disjunctivity and conjunctivity in the (Boolean) programming logic; *demonic* programs may contain 'worst-case' nondeterministic choices, and are characterised by conjunctivity alone; demonic/*angelic* programs may contain 'best-case' nondeterministic choice as well, and have been shown [1] to be characterised simply by monotonicity.

Probabilistic 'expectation transformers' can be similarly presented. We show here that deterministic (but probabilistic) programs have a predictable *distribution* of output behaviours; and their associated expectation transformers are characterised by 'linearity'. Demonic programs are made by demonic choices between deterministic ones; their expectation transformers are characterised by 'sublinearity'. And demonic/angelic programs are made by angelic choices between demonic ones, with their expectation transformers characterised by 'semi-sublinearity'.

In Sect. 2 we review Kozen's work and our recent extensions of it. Sections 3, 4 and 5 then build the classes of deterministic, demonic and demonic/angelic programs. And finally Sect. 6 makes the extension to infinite state spaces: in most cases the earlier results are unaffected, and we merely indicate how their proofs can be re-used here.

Throughout we use infix '.' to denote function application and '$:=$' for 'is defined to be'. We use A, B to denote standard (Boolean) predicates, often punning them with the (characteristic) $\{0,1\}$-valued functions of the state space commensurate with the context and to avoid the syntactic clutter of an explicit embedding function. In particular for states $s, s' \ldots$ we write $\{s, s' \ldots\}$ for the set containing those states, for the predicate holding only for those states and for the function 1 on those states and 0 elsewhere. We use α, β for general expectations and S for the underlying state space. Other notation is introduced as necessary.

2 Review of existing results

2.1 Introduction

Because Kozen [6, 7] studied only deterministic probabilistic programs – those containing neither demonic nor angelic choice – his operational model for sequential probabilistic programs consists essentially of Markov processes [3]: a program's execution from a given initial state is described by giving the resulting probability distribution over final states, and for example sequential composition is then a multiplication of appropriate Markov transition matrices.

Our extension [17] added demonic nondeterminism to that operational model in a simple way – programs are no longer Markov processes, but rather are a generalised form; and a compututions evolves by first making a demonic choice, in effect selecting one distribution from a number of possibilities, and then using the selected distribution to resolve the immediately following probabilistic choice. Thus programs become functions from initial state to *sets* of results, in this case to sets of discrete sub-probability distributions over a (finite) state space.

Adding nondeterminism specialises general powerdomain techniques, but with one novel aspect – namely that we impose (probabilistic) convexity,[1] which we need to allow a refinement relation between demonic nondeterministic choice and probabilistic choice.

For a logic of total correctness over such a model, one might concentrate (as for standard program logic) only on statements that hold with probability 1, and for example Rao's logic [20] is based on that idea. However such a logic loses most of the detail of the probabilistic information. To obtain a more discriminating logic Kozen instead used 'expectations' to investigate the final distributions. Briefly his idea is as follows.

Predicates embed into the space of real-valued functions of the state space by identifying *true* and *false* with the reals 1 and 0 respectively; the predicates become 'characteristic functions'. The probability of a program's establishing a postcondition is then the expected value[2] of the characteristic function over the distribution generated by executing the program: for program *Prog*, initial state s and postcondition *post* as a characteristic function we write that *wp*.*Prog*.*post*.s.

Once demonic choice is added, expectation transformers are still applicable, yet the interpretation of *wp*.*Prog*.*post*.s is then the 'greatest guaranteed' probability that *post* holds after execution from s.

[1] This is distinct from the term applied in the construction of the Plotkin powerdomain and is explained below.

[2] The expectation of a $\{0, 1\}$-valued function A is the probability that the event 'A evaluates to 1' occurs.

We now review both the operational and the expectation-transformer models of probabilistic demonic programs; we state the (partial) Galois connection that exists between them, and the consequences of it. In the expectation logic we use arithmetic operators, very often binary operators lifted pointwise to expectations, such as minimum $\sqcap$, maximum $\sqcup$ and p-averaging:

$$a \,{}_p\!\oplus\, b \;:=\; p \times a + (1-p) \times b\,.$$

We will see that those operators in particular correspond to demonic, angelic and probabilistic choice.

2.2 The operational model

The operational view of probabilitic programs is based on distributions over states.

Definition 2.1 *For finite state space S the space of* **discrete sub-probability distributions** $(\overline{S}, \sqsubseteq)$ *over S is defined as follows:*

$$\overline{S} \;:=\; \{F\colon S \to [0,1] \mid \textstyle\sum_{s\colon S} F.s \leq 1\}\,,$$
$$\text{where } F \sqsubseteq F' \;\; \textit{iff} \;\; (\forall s\colon S \cdot F.s \leq F'.s)\,.$$

Thus the distributions are functions from S into the unit interval $[0,1]$ that sum to no more than 1, and the order $\sqsubseteq$ is just $\leq$ extended pointwise.

Note that the distributions do not necessarily sum to 1 exactly; the 'deficit' represents the probability of nontermination.[3]

Now the Smyth powerdomain over $\overline{S}$ is characterised by non-empty up-closed sets of distributions: we say that a subset U of $\overline{S}$ is *up-closed* if whenever F is in U and $F \sqsubseteq F'$ then also F' is in U. Up-closure is one of our conditions on result sets.

Another condition is Cauchy closure:

Definition 2.2 *A set $A \subseteq \overline{S}$ is* **Cauchy-closed** *iff it is a closed subset of $\mathbb{R}^S$ considered as an $|S|$-dimensional space with the Euclidean topology, where $|S|$ denotes the cardinality of S.*

Cauchy closure will guarantee our continuity requirement – a recursive program is the limit of the nested result sets that represent its partial unfoldings, which limit must be nonempty. That is guaranteed by the fact[4] that the intersection of a nested family of nonempty closed sets of a compact topological

[3] An alternative though less convenient way to model nontermination is to introduce a special bottom state.

[4] We use the finite intersection lemma, a result from topology, which ensures that the intersection of a family of closed sets contained within a compact set is non-empty provided that the intersection of any finite subset of the family is non-empty.

space is nonempty; but we defer discussion of the compactness of $\overline{S}$ until Sect. 6.

Before we define our operational model we need one more closure condition – the novel *convexity* referred to above – that is not supplied by the standard Smyth construction: it is specific to the space of distributions.

Definition 2.3 *A set U in $\overline{S}$ is said to be* **convex** *if for any real $p \in [0,1]$ and for any F, F' in U we have also*

$$F \,{}_p\!\oplus F' \quad \in \quad U \;,$$

where the scalar operator ${}_p\!\oplus$ is lifted pointwise to real-valued functions: thus

$$(F \,{}_p\!\oplus F').s \quad = \quad p \times (F.s) + (1-p) \times (F'.s) \;.$$

Convex closure is imposed so that a demonic choice made between two programs is 'refined' (defined below) by a probabilistic choice, and we justify that by reasoning that a demonic choice is certainly no better (and in fact probably worse) than one determined by the flip of a coin: in the former case nothing about the method of choosing is known whereas in the latter the distribution is.[5]

Definition 2.4 *The space of* **probabilistic, demonic programs** $(\mathcal{H}S, \sqsubseteq)$ *is given by*

$$\mathcal{H}S \quad : = \quad S \to \mathcal{S}S \;,$$

where $\mathcal{S}S$ is the set of nonempty, up-closed, convex and Cauchy-closed subsets of $\overline{S}$, and the refinement order $\sqsubseteq$ is induced pointwise from the Smyth order on $\mathcal{S}S$. Thus for r, r' in $\mathcal{H}S$ we have

$$r \sqsubseteq r' \quad : = \quad (\forall s \colon S \cdot r.s \supseteq r'.s) \;.$$

That concludes the operational model; Figure 1 gives an example of a deterministic program.

2.3 The programming logic

We now turn to the probabilistic logic [17,7]. Denoting the non-negative reals by $\mathbb{R}_{\geq}$, we replace predicates (Boolean-valued functions of S) by *bounded expectations* of type $S \to \mathbb{R}_{\geq}$, denoted $\mathcal{P}S$, where by "bounded"

[5] This is consistent with an assumption that the probability distributions are unaffected by environmental conditions.

$$\begin{array}{l}\textbf{var}\ s\colon 1..N;\\ \textbf{do}\\ \quad \textbf{break}\ {}_{\frac{1}{s}}\!\oplus s := s-1\\ \textbf{od}\end{array}$$

This deterministic program takes each initial state s in the finite state space $1..N$ to a single distribution which is 'choose uniformly from $1..s$'. (The **do** $\cdots$ **od** construction means 'loop until **break** is executed'.)
We say 'deterministic' – although probabilistic nondeterminism is present – because there is no demonic or angelic nondeterminism in them; more importantly, such programs, like standard deterministic terminating programs, are maximal in the refinement order $\sqsubseteq$ (if miracles [11,18] are excluded).

Fig. 1. Deterministic program over finite state space

expectation we mean that there is some real number which it nowhere exceeds.[6] We relate expectations by *probabilistic implication* $\Rrightarrow$ defined (with its variants) over $\mathcal{P}S$ by

$$\begin{array}{ll}\Rrightarrow & \text{'everywhere no more than'}\\ \equiv & \text{'everywhere equal to'}\\ \Lleftarrow & \text{'everywhere no less than'}\ .\end{array}$$

We define the space of $\Rrightarrow$-monotonic expectation transformers as follows:

Definition 2.5 *The space of* **monotonic expectation transformers** $(\mathcal{T}S, \sqsubseteq)$ *is given by the set of monotone functions*

$$\mathcal{T}S \ :=\ \mathcal{P}S \to \mathcal{P}S\ ,$$

where for t, t' *in* $\mathcal{T}S$

$$t \sqsubseteq t' \quad \textit{iff} \quad (\forall \alpha : \mathcal{P}S \cdot t.\alpha \Rrightarrow t'.\alpha)\ .$$

The $\Rrightarrow$ symbol reminds us of Boolean implication $\Rightarrow$ since *false* corresponds to the everywhere 0 expectation, and *true* to everywhere 1, and just as *false* $\Rightarrow$ *true* we have $0 \Rrightarrow 1$.[7]

[6] For finite state space S the expectations are bounded trivially of course; but in Sect. 6, where we deal with infinite S, boundedness must be imposed.

[7] An immediate benefit is for example the conventional appearance of Definition 2.5 when compared with the definition of refinement between standard transformers t, t'

$$t \sqsubseteq t' \quad \text{iff} \quad (\forall A : \mathbb{P}S \cdot t.A \Rightarrow t'.A)\ ,$$

where $\mathbb{P}S$ is the set of predicates over S.

The expectation-transformer model generalises the predicate-transformer model for standard (non-probabilistic) demonic programs: programs take 'post-expectations' to 'greatest pre-expectations'. An informal introduction to the logic and its use is given elsewhere [14, 12]. As an example, note that the program of Fig. 1 – call it *ProgA* – satisfies $wp.ProgA.\{1\}.s = 1/s$, that is that the probability of final state $s = 1$ is just the reciprocal of the initial value of s.

2.4 The (partial) Galois connection

We now turn to the partial Galois connection between $\mathcal{H}S$ and $\mathcal{T}S$: it too generalises a standard construction. 'Relational programs' in $\mathcal{H}S$ can be embedded into the transformer space $\mathcal{T}S$, which is strictly richer; and 'transformer programs' can be projected back into the relational space, possibly moving up the $\sqsubseteq$-order as a result.

We begin with the embedding, a mapping from $\mathcal{H}S$ to $\mathcal{T}S$.

Definition 2.6 *Let h be a program in $\mathcal{H}S$, so taking initial states in S to sets of final distributions over S. Then the* **greatest pre-expectation** *at state s of program h, with respect to post-expectation α in $\mathcal{P}S$, is defined*

$$wp.h.\alpha.s \;\; := \;\; (\sqcap F : h.s \bullet \int_F \alpha) \,,$$

where $\int_F \alpha$ denotes the expected value[8] *of the real-valued function α of S over the distribution F on S, and $\sqcap$ selects the greatest lower bound. We call transformer $wp.h$ the* **embedding** *of h.*

Note from the definition that any upper bound for α is an upper bound for $wp.h.\alpha$ also, since $\int_F \alpha$ can be no more than $\sqcup\alpha$; thus boundedness is preserved by $wp.h$ even if S is infinite.

As discussed in the introduction, in the special case when an expectation A is $\{0,1\}$-valued, the pre-expectation $wp.r.A.s$ is the greatest guaranteed probability that A holds after execution of r from s – thus with that rationalisation we have replaced 'holds with certainty' (as in standard *wp*) by 'holds with probability'.

Now we describe the projection, taking expectation transformers in $\mathcal{T}S$ back to relational programs in $\mathcal{H}S$:

[8] In fact $\int_F \alpha$ is just $\sum_{s:\,S} \alpha.s \times F.s$ because S is finite and F is discrete [3]. Although the expression treats F and α symmetrically in the finite case (a dot product), in the infinite case even the types of F and α are different – and for that reason we use the $\int$-notation; it is also less cluttered.

Definition 2.7 *For expectation transformer t in $\mathcal{T}S$ the least program $rp.t$ in $\mathcal{H}S$ such that $t \sqsubseteq wp.(rp.t)$ is defined, if it exists, by*

$$rp.t.s \;:=\; \{F\colon \overline{S} \mid (\forall \alpha\colon \mathcal{P}S \cdot t.\alpha.s \leq \int_F \alpha)\}\ .$$

We call $rp.t$ the **projection** *of t.*

The first main result concerning rp and *wp* is that they establish a partial Galois correspondence between $\mathcal{H}S$ and $\mathcal{T}S$. We restate it here.

Theorem 2.8 [16. Lemma 8.2, 8.3] *$\mathcal{H}S$ and $\mathcal{T}S$ are put in partial Galois correspondence by rp and wp. Whenever rp is defined, the following inequations hold:*

$$\begin{array}{ll} id & \sqsubseteq wp \circ rp \\ rp \circ wp & = id\ , \end{array}$$

where id is the identity mapping in the respective spaces.

The partiality of rp in Definition 2.7 is caused by the possibility that the set

$$\{F\colon \overline{S} \mid (\forall \alpha\colon \mathcal{P}S \cdot t.\alpha.s \leq \int_F \alpha)\}$$

might be empty, and we have excluded for this presentation empty results sets from $\mathcal{H}S$: they correspond to 'miracles' [11,18]. An alternative would be to allow miracles, and the Galois connection would become total; we would then adjoin ∞ to $\mathbb{R}_{\geq}$, to make it and $\mathcal{P}S$ become $\sqcup$-complete.[9]

The importance of Theorem 2.8 lies in its corollary, and it forms the second main result. That result establishes 'healthiness' conditions in the style of Dijkstra which identify the *wp*-images of $\mathcal{H}S$: they identify the subset of 'implementable' programs embedded within $\mathcal{T}S$. The result makes clear when theorems proved in $\mathcal{T}S$ correspond to theorems about programs in $\mathcal{H}S$, which is important since (as for standard predicate transformers) $\mathcal{T}S$ is a more convenient framework for calculation than $\mathcal{H}S$ is.

The healthiness conditions may be described as three properties of expectation transformers which together characterise the *wp*-images of programs (Theorem 2.9 below). For a non-negative real a we define scalar multiplication lifted to expectations (for example $a \times \alpha$ in (1)) by pointwise multiplication; similarly we lift the binary operator $\ominus$ on reals defined $a \ominus b\colon = (a - b) \sqcup 0$, where $\sqcup$ is the binary maximum operator; and for a

[9] We do not make those extensions here because ∞ would complicate the arithmetic in our principal use of the Galois connection in this report, which is to establish healthiness conditions for the subspaces of $\mathcal{T}S$.

real a we denote by $\underline{a}$ the constant expectation everywhere evaluating to a. The three properties are then

$$\begin{array}{lll} t.(a \times \alpha) & \equiv a \times t.\alpha & \text{scaling} \\ t.(\alpha + \alpha') & \Leftarrow t.\alpha + t.\alpha' & \text{sub-additivity} \\ t.(\alpha \ominus \underline{a}) & \Leftarrow t.\alpha \ominus \underline{a} & \ominus\text{-subdistribution} . \end{array} \tag{1}$$

We refer collectively to the properties (1) as 'sublinearity'.[10]

The scaling property is particular to an expectation model of probability: if an expectation is scaled by a constant a then the expected value with respect to a distribution is similarly scaled. The two subdistribution properties, on the other hand, together generalise standard conjunctivity [17, following Lem. 7.2].

The importance of sublinearity is that it exactly characterises *wp*-images:

Theorem 2.9 *[17, Thm. 8.7] Given any t in $\mathcal{T}S$, there is an r in $\mathcal{H}S$ with $t = wp.r$ exactly when t is sublinear (1).*

We denote the *wp*-images of $\mathcal{H}S$, or equivalently by Theorem 2.9 the subset of sublinear transformers, by $\mathcal{T}_{\sqcap}\, S$; we shall call them *demonic* expectation transformers,[11] and we return to them in Sect. 4.

That concludes the review of our earlier treatment [17] of demonic probabilistic programs over finite state spaces.

3 Deterministic programs

Standard deterministic programs deliver a single output state for any fixed input; probabilistic deterministic programs deliver a single output *distribution* for any fixed input. (Recall Fig. 1.) Thus even though separate runs of a deterministic, probabilistic program may deliver different outputs from the same intial state, over a group of runs a tabulation of results will reveal a probability distribution of those outputs; and because the program is deterministic, a second group of runs from the same initial state will reveal the same distribution.

Since standard deterministic programs are disjunctive as well as conjunctive – and those properties are $(\neg)$-duals – we conjecture that probabilistic

[10] These properties can be more concisely (and correspondingly more cryptically) expressed as follows [17, Def. 7.1]: expectation transformer t is sublinear if and only if for all non-negative scalars a, b, c and expectations α, β we have

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

Note that $+$ binds more tightly than $\ominus$.

[11] Earlier [17] we called these the 'regular' transformers, because they correspond to relations.

programs are *super*- as well as sub-additive, since those properties are $(-)$-duals: that is, deterministic programs are simply additive. Our main result in this section is to show that additivity, scaling and $\ominus$-subdistribution (referred to collectively as 'linearity' below) indeed characterise determinism.

We begin by fixing the idea of a deterministic operational program – it is one that is characterised by a single output distribution.

Definition 3.1 *A program h in $\mathcal{H}S$ is* **deterministic** *iff for all states s there is a single distribution F_s so that*

$$h.s \;:=\; \{F\colon \overline{S} \mid F_s \sqsubseteq F\}\;.$$

Notice how nonterminating programs are also covered by the definition. Deterministic here means "deterministic if terminating" and is thus closer to the notion of pre-deterministic used by some authors [4].

Next we define linearity.

Definition 3.2 *An expectation transformer t is said to be* **linear** *iff it is $\ominus$-subdistributive, scaling and* **additive***, where for that last we mean that for all expectations α, α' in $\mathcal{P}S$ we have*

$$t.(\alpha+\alpha').s \;=\; t.\alpha.s + t.\alpha'.s\;.$$

The proof that deterministic programs map to linear expectation transformers under *wp* is straightforward.

Lemma 3.3 *If h in $\mathcal{H}S$ is deterministic then wp.h is linear.*

Proof. Theorem 2.9 ensures that *wp*.h is sublinear at least; then its additivity follows directly from Definition 3.1 and the additivity of $\int_F$ (equivalently the additivity of expectation in elementary probability [3]). □

More involved is the converse, that linearity implies determinism.

Lemma 3.4 *If a transformer t in $\mathcal{T}S$ is linear, then it is the wp-image of a deterministic relational program.*

Proof. For expectation transformer t in $\mathcal{T}S$ we define the deterministic program h in $\mathcal{H}S$ as follows:

$$\begin{aligned} h.s \;&:=\; \{F\colon \overline{S} \mid F_s \sqsubseteq F\} \\ \text{where } F_s.s' \;&:=\; t.\{s'\}.s\;, \end{aligned}$$

writing $\{s'\}$ for the 'point' expectation that is 1 at s' and 0 elsewhere.

Observe that h is well defined: F_s is indeed an element of $\overline{S}$ because

Let *ProgB* be the deterministic program

$$\begin{array}{l}\textbf{var}\ s\colon \{\mathit{Heads}, \mathit{Tails}\};\\ s := \mathit{Heads}\ {}_{\frac{1}{2}}\!\oplus\ s := \mathit{Tails}\ .\end{array}$$

We have $wp.\mathit{ProgB}.\{\mathit{Heads}\}.s = 1/2$ for all initial values of s, and similarly $wp.\mathit{ProgB}.\{\mathit{Tails}\}.s = 1/2$. Now (punning sets and characteristic functions) we have $\{\mathit{Heads}\} + \{\mathit{Tails}\} = \{\mathit{Heads}, \mathit{Tails}\}$, and

$$wp.\mathit{ProgB}.\{\mathit{Heads}, \mathit{Tails}\}.s \quad = \quad 1 \quad = \quad 1/2 + 1/2\ ,$$

thus illustrating additivity.

Fig. 2. Deterministic programs are additive

$$\begin{array}{lll}
 & \textstyle\sum_{s'\colon S} F_s.s' & \\
= & t.(\textstyle\sum_{s'\colon S}\{s'\}).s & t \text{ additive; definition } F\\
= & t.\underline{1}.s & \text{definition } \underline{1}\\
\leq & (t.\underline{1}.s \ominus 1) + 1 & \text{arithmetic}\\
\leq & t.\underline{0}.s + 1 & t \text{ is } \ominus\text{-subdistributive}\\
= & 0 \times t.\underline{0}.s + 1 & t \text{ scaling}\\
= & 1\ . &
\end{array}$$

We now show that in fact $t = wp.h$. Observe first that any expectation α in $\mathcal{P}S$ is a (finite) sum $\sum_{s:S} (\alpha.s) \times \{s\}$ of characteristic functions. (The sum is finite because S is.) We reason

$$\begin{array}{lll}
 & t.\alpha.s & \\
= & t.(\textstyle\sum_{s':S}\ \alpha.s' \times \{s'\}).s & \\
= & \textstyle\sum_{s':S}\ \alpha.s' \times t.\{s'\}.s & t \text{ linear}\\
= & \textstyle\sum_{s':S}\ \alpha.s' \times F_s.s' & \text{definition of } F_s\\
= & \textstyle\int_{F_s}\ \alpha & \text{definition of } \textstyle\int\\
= & wp.h.\alpha.s\ . & \textstyle\int \text{ monotonic; Lemma 3.4 of } h\text{; Definition 2.6}
\end{array}$$

□

We call the space of linear – equivalently deterministic – transformers $\mathcal{T}_\circ\, S$, and have our theorem:

Theorem 3.5 *A transformer is linear iff it is the wp-image of a deterministic relational program.*

Proof. Lemma 3.3, 3.4. Figure 2 illustrates the result. □

4 Demonic programs

Demonic programs are constructed by closing $\mathcal{T}_\circ\, S$ under demonic choice $\sqcap$: we show here that they are exactly the sublinear transformers $\mathcal{T}_\sqcap\, S$ described

var s: $1..N$;

do $s \neq 1 \to$ /* choose s demonically from $1..s$ */
  break $\sqcap\, s := s - 1$
od;

do /* choose s uniformly from $1..s$ */
  break $\frac{1}{s}\oplus\, s := s - 1$
od

This demonic program takes an initial state s in $1..N$ to any 'anti-monotonic' distribution over $1..s$ that assigns higher probabilities to lower outcomes.

Fig. 3. Demonic program over finite state space

in Sect. 2, which were the subject of our earlier work [17]. Figure 3 shows a 'typical' demonic program: first natural number s is decreased demonically from its initial value, but not below 1; then a uniform probabilistic choice is made from the numbers $1..s$. The final distribution over a set of runs is not (wholly) predictable, because of the nondeterminism: in separate sets one could find quite different distributions. But each distribution will have the property that higher frequencies are associated with lower final states; and it can be shown that the program is capable of producing all such 'anti-monotonic' distributions.

To establish our characterisation of $\mathcal{T}_\sqcap\, S$, we note first from simple arithmetic that if t and t' are additive then $t \sqcap t'$ is sub-additive (by subdistribution of greatest lower bounds through addition), and that scaling and $\ominus$-subdistribution are preserved. Thus the elements of $\sqcap$-closure of $\mathcal{T}_\circ\, S$ are sublinear.

For the converse, we note that if t is sublinear then from Theorem 2.9 and Theorem 2.8 we have $wp.(rp.t) = t$;[12] thus take for the deterministic 'components' of t the programs in $\mathcal{T}_\circ\, S$ generated (Definition 3.1) by choosing F_s in all possible ways from $rp.t.s$ for each s. The result is then a consequence of the correspondence between $\cup$ in $\mathcal{H}S$ and $\sqcap$ in $\mathcal{T}S$. Thus any sublinear transformer may be written as the (possibly infinite) demonic choice $\sqcap$ between members of $\mathcal{T}_\circ\, S$, and we have proved this theorem:

Theorem 4.1 *The $\sqcap$-closure of the deterministic transformers $\mathcal{T}_\circ\, S$ is exactly the sublinear transformers $\mathcal{T}_\sqcap\, S$.*

Figure 4 gives an illustration of sublinearity.

[12] This is our principal use of the Galois connection: note that sublinearity of t implies it is not miraculous [17], and that in turn ensures definedness of rp.

Let *ProgC* be the program

$$\begin{array}{l}\textbf{var}\ s\colon \{\mathit{Heads}, \mathit{Tails}\};\\ s := \mathit{Heads} \ \sqcap\ s := \mathit{Heads} \ {}_{\frac{2}{3}}\!\oplus\ s := \mathit{Tails}\ .\end{array}$$

This program contains both probabilistic and demonic choice; it chooses between *Heads* and *Tails*, assigning a probability of *at least* $2/3$ to the former. Now we have $wp.\mathit{ProgC}.\{\mathit{Heads}\}.s = 2/3$ because the nondeterminism is demonic (not angelic) – although *Heads* can be chosen always, we can be *sure* it will be chosen only two-thirds of the time. Similarly $wp.\mathit{ProgC}.\{\mathit{Tails}\}.s = 0$ because *Tails* might never be chosen at all. Now

$$wp.\mathit{ProgC}.\{\mathit{Heads}, \mathit{Tails}\}.s \quad = \quad 1 \quad > \quad 2/3 + 0\ ,$$

thus illustrating sublinearity and the failure of additivity.

Fig. 4. Demonic programs are sublinear but not additive

5 Angelic/demonic programs

Our final step is to move from $\mathcal{T}_{\sqcap}\, S$ to the more general space formed by allowing angelic choice as well. In the standard case one drops conjunctivity; here we will drop sub-additivity.

Our generalisation of angelic choice in the probabilistic domain is as follows:

Definition 5.1 *For expectation transformers t, t' in $\mathcal{T}S$, expectation α in $\mathcal{P}S$ and state s in S we define*

$$(t \sqcup t').\alpha.s \quad : = \quad t.\alpha.s \ \sqcup\ t'.\alpha.s\ .$$

We denote the closure of $\mathcal{T}_{\sqcap}\, S$ under $\sqcup$ by $\mathcal{T}_{\sqcap\!\sqcup}\, S$ (using $\sqcup$ for least upper bound in the semantics); its characterisation will turn out to be 'semi-sublinearity'.

Definition 5.2 *Expectation transformer t in $\mathcal{T}\,S$ is said to be* semi-sublinear *iff it satisfies the properties of scaling and $\ominus$-subdistribution but not (necessarily) sub-additivity:*

$$\begin{array}{ll} t.(a \times \alpha) \equiv a \times (t.\alpha) & \textit{scaling}\\ t.(\alpha \ominus \underline{a}) \Leftarrow t.\alpha \ominus \underline{a} & \ominus\textit{-subdistribution}\ .\end{array}$$

We show first that all members of $\mathcal{T}_{\sqcap\!\sqcup}\, S$ are semi-sublinear.

Lemma 5.3 *If t is in $\mathcal{T}_{\sqcap\!\sqcup}\, S$ then it is semi-sublinear.*

Proof. We need only show that semi-sublinearity is preserved by (possibly infinite) applications of $\sqcup$, and that is immediate from arithmetic and Definition 5.2. □

Our major goal however is to show that $\mathcal{T}_{\sqcap\!\!\sqcap} S$ comprises *exactly* the semi-sublinear transformers, and for that we need the converse of Lemma 5.3: that all semi-sublinear transformers in $\mathcal{T}S$ are members of $\mathcal{T}_{\sqcap\!\!\sqcap} S$.

Motivated by Back and von Wright's construction [1, Lemma 8] for standard programs, given a semi-sublinear t we consider all post-expectations β and express t as the angelic choice over a family t_β of transformers in $\mathcal{T}_\sqcap S$. Each transformer t_β will be the $\sqsubseteq$-weakest transformer that itself is monotonic and semi-sublinear[13] and agrees with t at β: because t_β is the weakest such, it will be everywhere no more than t; and because each t_β agrees with t at β, their supremum will be at least t. The surprise will be that the t_β's turn out to be fully (rather than just semi-) sublinear, making them members of $\mathcal{T}_\sqcap S$ as required.

The appropriate definition of t_β is the following:

Definition 5.4 *For t a semi-sublinear transformer in $\mathcal{T}S$ and expectation β in $\mathcal{P}S$, let the* β-component t_β *of t be defined by*

$$t_\beta.\alpha.s \;:=\; (\sqcup c, c' : \mathbb{R}_\geq \mid c \times \beta - \underline{c'} \Rrightarrow \alpha \cdot c \times (t.\beta.s) - c') \;,$$

for α in $\mathcal{P}S$ and s in S.

To see that t_β is well defined, by which we mean that when applied to α in $\mathcal{P}S$, the result is nowhere infinite nor negative, we appeal first to Lemma 5.6 below to see that $t_\beta.\alpha$ must be everywhere finite; to see that the result is non-negative we need only put c, c' to be $0, 0$ in Definition 5.4 and then notice that the definition maximises.

We now prove the three crucial properties of β-components.

The first property is that t_β is no less than t at β.

Lemma 5.5 *For all semi-sublinear transformers t and expectations β we have $t.\beta \Rrightarrow t_\beta.\beta$.*

Proof. Take $c, c' := 1, 0$ in Definition 5.4. □

The second property is that t_β is no more than t in general.

Lemma 5.6 *For all semi-sublinear transformers t and expectations β we have $t_\beta \sqsubseteq t$.*

Proof. Take arbitrary expectation α in $\mathcal{P}S$ and state s. For all c, c' in $\mathbb{R}_\geq$,

[13] The impact of requiring semi-sublinearity in t_β is that the otherwise obvious choice

$$t_\beta.\alpha \;:=\; t.\beta \text{ if } (\beta \Rrightarrow \alpha) \text{ else } 0 \qquad (2)$$

is not allowed – by scaling, for example, we see that $t_\beta.(2\beta)$ must be $2(t.\beta)$ rather than the $t.\beta$ that (2) would give. Those properties of t_β motivate Definition 5.4 that makes it monotonic and semi-sublinear by construction, leaving the job of Lemma 5.7 mainly to establish its sub-additivity.

$$
\begin{array}{ll}
\quad c \times \beta - \underline{c'} \Rrightarrow \alpha & \\
\text{implies} c \times \beta \ominus \underline{c'} \Rrightarrow \alpha & \underline{0} \Rrightarrow \alpha \\
\text{implies} t.(c \times \beta \ominus \underline{c'}).s \leq t.\alpha.s & t \text{ monotonic} \\
\text{implies} c \times (t.\beta.s) \ominus c' \leq t.\alpha.s & t \text{ semi-sublinear} \\
\text{implies} c \times (t.\beta.s) - c' \leq t.\alpha.s\ . & \text{arithmetic}
\end{array}
$$

Since the inequality holds for arbitrary c, c' it holds for the supremum taken over them in Definition 5.4. □

Finally we show sublinearity of the β-components.

Lemma 5.7 *For semi-sublinear t in $\mathcal{T}S$ and β in $\mathcal{P}S$ the component t_β is sublinear.*

Proof. The scaling of t_β is obvious from its definition. The $\ominus$-subdistributivity is straightforward also:

$$
\begin{array}{ll}
\quad & t_\beta.(\alpha \ominus \underline{a}).s \\
= & (\sqcup c, c' \colon \mathbb{R}_{\geq} \mid c \times \beta - \underline{c'} \Rrightarrow \alpha \ominus \underline{a} \cdot c \times (t.\beta.s) - c') \quad \text{Definition 5.4} \\
\geq & \qquad\qquad \text{take } c' := c'' + a \\
& (\sqcup c, c'' \colon \mathbb{R}_{\geq} \mid c \times \beta - (\underline{c'' + a}) \Rrightarrow \alpha \ominus \underline{a} \cdot c \times (t.\beta.s) - (c'' + a)) \\
\geq & (\sqcup c, c'' \colon \mathbb{R}_{\geq} \mid c \times \beta - \underline{c''} \Rrightarrow \alpha \cdot (c \times (t.\beta.s) - c'') - a) \quad \text{arithmetic} \\
\geq & (\sqcup c, c'' \colon \mathbb{R}_{\geq} \mid c \times \beta - \underline{c''} \Rrightarrow \alpha \cdot (c \times (t.\beta.s) - c'')) - a \quad \text{arithmetic} \\
= & t_\beta.\alpha.s - a\ , \qquad\qquad \text{Definition 5.4}
\end{array}
$$

and hence, since $t_\beta.(\alpha \ominus \underline{a})$ is everywhere non-negative, we can deduce that indeed

$$t_\beta.(\alpha \ominus \underline{a}).s \geq (t_\beta.\alpha \ominus \underline{a}).s\ .$$

The main point however is sub-additivity, for which we argue

$$
\begin{array}{ll}
\quad & t_\beta.(\alpha_1 + \alpha_2).s \\
= & (\sqcup c, c' \colon \mathbb{R}_{\geq} \mid c \times \beta - \underline{c'} \Rrightarrow \alpha_1 + \alpha_2 \cdot c \times (t.\beta.s) - c') \quad \text{Definition 5.4} \\
\\
= & (\sqcup c_1, c_2, c_1', c_2' \colon \mathbb{R}_{\geq} \qquad \text{take } c, c' := c_1 + c_2, c_1' + c_2' \\
& \quad \mid (c_1 + c_2)\beta - (\underline{c_1' + c_2'}) \Rrightarrow \alpha_1 + \alpha_2 \\
& \quad \cdot (c_1 + c_2)(t.\beta.s) - (c_1' + c_2')) \\
\\
\geq & (\sqcup c_1, c_2, c_1', c_2' \colon \mathbb{R}_{\geq} \qquad \text{arithmetic} \\
& \quad \mid c_1 \times \beta - \underline{c_1'} \Rrightarrow \alpha_1 \ \wedge \ c_2\beta - \underline{c_2'} \Rrightarrow \alpha_2 \\
& \quad \cdot (c_1 \times (t.\beta.s) - c_1') + (c_2 \times (t.\beta.s) - c_2')) \\
\\
= & \qquad\qquad c_1, c_1' \text{ and } c_2, c_2' \text{ vary independently} \\
& \quad (\sqcup c_1, c_1' \colon \mathbb{R}_{\geq} \mid c_1 \times \beta - \underline{c_1'} \Rrightarrow \alpha_1 \cdot c_1 \times (t.\beta.s) - c_1') \\
& + (\sqcup c_2, c_2' \colon \mathbb{R}_{\geq} \mid c_2 \times \beta - \underline{c_2'} \Rrightarrow \alpha_2 \cdot c_2 \times (t.\beta.s) - c_2')
\end{array}
$$

Let *ProgD* be the program

$$\begin{array}{l}\textbf{var}\ s\colon \{\mathit{Heads}, \mathit{Tails}\};\\ s := \mathit{Heads}\ {}_{\frac{1}{3}}\!\oplus\ s := \mathit{Tails}\ \ \sqcap\ \ s := \mathit{Tails}\ ,\end{array}$$

and take *ProgC* as in Fig. 4. Programs *ProgC* and *ProgD* are in $\mathcal{T}_{\sqcap}\, S$; let *ProgE* in $\mathcal{T}_{\boxplus}\, S$ be the angelic choice *ProgC* $\sqcup$ *ProgD* between them.
Program *ProgE* (confusingly) chooses *angelically* whether to use a *Heads*-biased or *Tails*-biased coin, and then applies the chosen bias demonically.
Now we have $\mathit{wp}.\mathit{ProgE}.\{\mathit{Heads}\}.s = 2/3 \sqcup 0 = 2/3$ because the nondeterminism is angelic between the biases, and once the *Heads* bias is chosen, the worst it can do is to select *Heads* only two-thirds of the time. Similarly $\mathit{wp}.\mathit{ProgE}.\{\mathit{Tails}\}.s = 2/3$. Yet

$$\mathit{wp}.\mathit{ProgE}.\{\mathit{Heads}, \mathit{Tails}\}.s \quad = \quad 1 \quad < \quad 2/3 + 2/3\ ,$$

thus illustrating the failure of sublinearity.

Fig. 5. Angelic programs are not sublinear

$$= \qquad t_\beta.\alpha_1.s + t_\beta.\alpha_2.s\ . \qquad \text{Definition 5.4}$$

□

That gives our main theorem for this section:

Theorem 5.8 *The $\sqcup$-closure of the demonic transformers $\mathcal{T}_{\sqcap}\, S$ is exactly the semi-sublinear transformers $\mathcal{T}_{\boxplus}\, S$.*

Proof. For semi-sublinear expectation transformer t and expectation α in $\mathcal{P}S$ we have

$$\begin{array}{lll} & t.\alpha & \\ \Rrightarrow & t_\alpha.\alpha & \text{Lemma 5.5}\\ \Rrightarrow & (\sqcup\beta\colon \mathcal{P}S \cdot t_\beta).\alpha & \\ \Rrightarrow & t.\alpha\ . & \text{Lemma 5.6}\end{array}$$

Thus we have $t = (\sqcup\beta\colon \mathcal{P}S \cdot t_\beta)$, and conclude from Lemma 5.7 that indeed t is the supremum of sublinear transformers.
The converse implication is given by Lemma 5.3. □

Figures 5 and 6 illustrate the behaviour of transformers in $\mathcal{T}_{\boxplus}\, S$.

6 Infinite state spaces

Our earlier results about $\mathcal{H}S$ and $\mathcal{T}S$ [17] were summarised in Sect. 2, and assumed that the state space is finite; the subsequent analysis of Sections 3–5 also assumed finiteness. We now show how to extend all those results to infinite state spaces; the principal tool will be continuity.

Let *ProgF* be the program

$$\textbf{skip} \ {}_{\frac{2}{3}}\oplus \ \textbf{abort} \ .$$

This deterministic program takes any initial state s to the final distribution assigning $2/3$ to s and 0 to all other states. The 'deficit' $1-2/3$ is the probability $1/3$ of nontermination caused by the primitive non-terminating program **abort**. Now have *wp.ProgF*$.1.s = 2/3$ for all s, and indeed *wp.ProgF*$.r.s = 2r/3$ for any real number r, illustrating scaling. An illustration of $\ominus$-subdistribution is that

$$\textit{wp.ProgF}.(1 \ominus 1/2).s \quad = \quad 2/3 \times (1 \ominus 1/2) \quad > \quad 1/6 \quad = \quad 2/3 \ominus 1/2 \ .$$

Fig. 6. Angelic programs are semi-sublinear

In summary there will be two changes: the first is that we must explicitly restrict expectations to $\mathcal{P}_B S$, the subset of $\mathcal{P}S$ consisting of the *bounded* functions from S into $\mathbb{R}_{\geq}$; and the second is that we must address the issue of continuity explicitly.[14] The restriction to bounded expectations α is necessary to guarantee that $\int_F \alpha$ is well defined (is finite in value)[15]. And continuity of *wp.Prog* for *Prog* in $\mathcal{H}S$ is no longer to be taken for granted: its derivation from sublinearity [17, Lem. 7.6] depended on finiteness of S. Thus when S is infinite we must prove continuity directly from Definition 2.4 (the operational model) and Definition 2.6 (*wp*); and that forms the main technical contribution of this section.

Once continuity is dealt with, most of our earlier results [17] can be re-established by appeal to their proofs for the finite case.

We begin by introducing some technical details of the probability distributions over the now-infinite state space.

6.1 Topological preliminaries; compactness of $\overline{S}$

Our set of distributions $\overline{S}$ is a subset of the function space $S \to \mathbb{R}_{\geq}$, which in turn can be regarded equivalently as Euclidean space $\mathbb{R}_{\geq}^{S}$ but now possibly of infinite dimension.[16] The principal notions we use there are topological clo-

[14] Finiteness of S gives boundedness and continuity automatically; so these issues do not limit our earlier results in any way.

[15] In the infinite case we define $\int_F \alpha$ by

$$(\sqcup T : \mathbb{F}S \cdot \sum_{s:\, T} F.s \times \alpha.s) \ .$$

[16] One generalisation we do not (need to) make is from discrete distributions over finite S to measures over an infinite S. That is because our programs' only access to probability is via a binary operator ${}_p\oplus$, and they therefore can generate only discrete result distributions, even when S is infinite.

sure and compactness, the latter being essential for establishing continuity of *wp*-images.

The *Euclidean topology* for the real line $\mathbb{R}_{\geq}$ corresponds to the usual Euclidean metric, and the topology $\mathcal{E}_S$ we use for $\mathbb{R}_{\geq}^S$ is the product of the Euclidean topologies over $\mathbb{R}_{\geq}$ for each dimension in S. A basis for $\mathcal{E}_S$ is then the collection of sets of the form

$$N \times \mathbb{R}_{\geq}^{S-T} \tag{3}$$

(ignoring order in the product), for all finite subsets T of S and open subsets N of $\mathbb{R}_{\geq}^T$. For dimension s in S we say that s is in the *support* of an open set in $\mathcal{E}_S$ just when the projection of that set onto s is not all of $\mathbb{R}_{\geq}$. Similar notions of finiteness apply to expectations:

Definition 6.1 *An expectation α in $\mathcal{P}S$ is* **finitary** *if there is a finite subset T of S such that $\alpha.s \neq 0$ only for states s in T. (Note that such expectations are bounded by construction.)*

The **support** *of an expectation α is the set of states s such that $\alpha.s \neq 0$; thus an expectation is finitary iff it has finite support.*

We write $\mathbb{F}S$ for the set of finite subsets of S, and $\mathcal{P}_f S$ for the set of finitary expectations over S.

For subset T of S, we define the **restriction** *of α to T as*

$$(\alpha{\downarrow}T).s \;:=\; \begin{array}{ll} \alpha.s & \textit{if } s \in T \\ 0 & \textit{otherwise.} \end{array}$$

Thus $\alpha{\downarrow}T$ is finitary if T is finite.

Our technique for extending our results will be to show that they continue to hold in infinite S provided we restrict our attention to finitary expectations; then continuity of the transformers involved will carry the equalities through to all expectations, since any expectation can be written as a directed $\sqcup$-limit of its finitary restrictions.

First we must investigate some properties of finitary expectations themselves; we begin by showing that finitary expectations can be used to define 'closed half-spaces' in $\mathbb{R}_{\geq}^S$.

Lemma 6.2 *For (finitary) α in $\mathcal{P}_f S$ and r in $\mathbb{R}_{\geq}$, sets of the forms*

$$\{F\colon \mathbb{R}_{\geq}^S \mid \int_F \alpha \leq r\} \quad \textit{and} \quad \{F\colon \mathbb{R}_{\geq}^S \mid \int_F \alpha \geq r\}$$

are closed in the topology $\mathcal{E}_S$ over $\mathbb{R}_{\geq}^S$.

Proof. Let the support of α be T, finite because α is finitary. Then in either case above the projection of the set is all of $\mathbb{R}_{\geq}$ for each dimension outside of T; and its projection into $\mathbb{R}_{\geq}^T$ is the complement of an open set there. □

For infinitary expectations, however, we have closure in one direction only.

Lemma 6.3 *For any expectation α in $\mathcal{P}_B S$ and r in $\mathbb{R}_{\geq}$, the set of distributions*

$$\{F\colon \mathbb{R}^S_{\geq} \mid \int_F \alpha \leq r\}$$

is closed in $\mathbb{R}^S_{\geq}$.

Proof. We have

$$\begin{array}{ll} & \{F\colon \mathbb{R}^S_{\geq} \mid \textstyle\int_F \alpha \leq r\} \\ = & \{F\colon \mathbb{R}^S_{\geq} \mid \textstyle\int_F (\sqcup T\colon \mathbb{F}S \cdot \alpha{\downarrow}T) \leq r\} \\ = & \qquad\qquad \text{bounded monotone convergence for } \textstyle\int_F \text{ [5]} \\ & \{F\colon \mathbb{R}^S_{\geq} \mid (\sqcup T\colon \mathbb{F}S \cdot \textstyle\int_F \alpha{\downarrow}T) \leq r\} \\ = & \{F\colon \mathbb{R}^S_{\geq} \mid (\forall T\colon \mathbb{F}S \cdot \textstyle\int_F \alpha{\downarrow}T \leq r)\} \\ = & (\cap T\colon \mathbb{F}S \cdot \{F\colon \mathbb{R}^S_{\geq} \mid \textstyle\int_F \alpha{\downarrow}T \leq r\})\,, \end{array}$$

which is an intersection of a (T-indexed) collection of closed sets (Lemma 6.2), since $\alpha{\downarrow}T$ is finitary for all T in $\mathbb{F}S$. □

That Lemma 6.3 works in only one direction is because our expectations (finitary or not) take only non-negative values. To see that its dual does not hold in general, consider the half-space

$$\{F\colon \mathbb{R}^S_{\geq} \mid \int_F \underline{1} \geq 1\}\,. \tag{4}$$

The origin (F everywhere 0) does not lie within it, yet every open set in the basis (3) containing the origin also intersects the half-space – for example, take F in $\mathbb{R}^S_{\geq}$ to be 1 at some point in $S{-}T$ and 0 elsewhere. Thus the origin is a limit point of (4) but is not in it.

With the above we now have the compactness property of our space of distributions referred to earlier.

Lemma 6.4 *The space $\overline{S}$ of distributions over S is a compact subset of $(\mathbb{R}^S_{\geq}, \mathcal{E}_S)$.*

Proof. The set $\overline{S}$ may be written

$$[0,1]^S \cap \{F\colon \mathbb{R}^S_{\geq} \mid \int_F \underline{1} \leq 1\}\,.$$

But $[0,1]^S$ is compact by Tychonoff's Theorem[17], and $\{F\colon \mathbb{R}^S_{\geq} \mid \int_F \underline{1} \leq 1\}$ is closed by Lemma 6.3. Thus $\overline{S}$ is the intersection of a compact set and a closed set, and is therefore compact. (It is also closed.) □

[17] *Tychonoff's Theorem* states that a product of compact sets is compact in the product topology.

6.2 The Galois functions; continuity

We now return to the connection (Theorem 2.8) between the relational space $\mathcal{H}S$ and the expectation-transformer space $\mathcal{T}S$: they are connected by the functions

$$\begin{array}{rll} rp\colon & \mathcal{T}S \to \mathcal{H}S & \text{(Definition 2.7)} \\ \text{and } wp\colon & \mathcal{H}S \to \mathcal{T}S & \text{(Definition 2.6)}\ . \end{array}$$

For well-definedness of rp in the finite case we showed [17] that given any t in $\mathcal{T}S$ (for which rp is defined) and s in S the set $rp.t.s$ was up-closed, convex and Cauchy-closed: and they are just the conditions placed on result sets in $\mathcal{H}S$ by Definition 2.4.

In the infinite case rp is not so well behaved: up-closure and convexity follow as before; but Cauchy-closure now requires an explicit appeal to the continuity of t. Here we are concerned with finite results, thus we use the weaker notion of 'bounded continuity'; it is also exactly the condition we need to establish our results.

Definition 6.5 *A transformer t in $\mathcal{T}S$ is* boundedly continuous *if for any directed subset $\mathcal{A}$ of expectations bounded above by some constant,*

$$t.(\sqcup\mathcal{A}) \quad = \quad (\sqcup\alpha\colon \mathcal{A}\cdot t.\alpha)\ .$$

The following technical lemma shows that when t is boundedly continuous we may restrict ourselves to finitary expectations in the use of Definition 2.7:

Lemma 6.6 *For any boundedly-continuous t in $\mathcal{T}S$ and s in S we have*

$$rp.t.s \quad = \quad \{F\colon \overline{S}\cdot(\forall\alpha\colon \mathcal{P}_fS\cdot t.\alpha.s \le \int_F \alpha)\}\ .$$

The only change from Definition 2.7 is the type of the universally quantified α – it is now drawn from $\mathcal{P}_fS$ rather than $\mathcal{P}S$.

Proof. For arbitrary α in $\mathcal{P}_BS$,

$$\begin{array}{ll} & (\forall T\colon \mathbb{F}S\cdot t.(\alpha{\downarrow}T).s \le \int_F \alpha{\downarrow}T) \\ \text{implies} & (\sqcup T\colon \mathbb{F}S\cdot t.(\alpha{\downarrow}T).s) \le (\sqcup T\colon \mathbb{F}S\cdot\ \int_F \alpha{\downarrow}T) \end{array}$$

iff $\qquad$ t boundedly continuous; bounded monotone convergence

$$t.(\sqcup T\colon \mathbb{F}S\cdot \alpha{\downarrow}T).s \le \int_F(\sqcup T\colon \mathbb{F}S\cdot \alpha{\downarrow}T)$$

iff $\qquad t.\alpha.s \le \int_F \alpha\ .$

The result then follows directly from Definition 2.7, since any $\alpha{\downarrow}T$ is finitary.

□

With Lemma 6.6 we can establish Cauchy-closure of $rp.t$ for boundedly-continuous t.

Lemma 6.7 *For any boundedly-continuous t in $\mathcal{T}S$ and state s in S, the set of distributions $rp.t.s$ is Cauchy-closed in $\overline{S}$.*

Proof. We reason

$$\begin{array}{lll} & rp.t.s & \\ = & \{F\colon \overline{S}\cdot (\forall \alpha\colon \mathcal{P}_f S\cdot t.\alpha.s \le \int_F \alpha)\} & \text{Lemma 6.6} \\ = & (\cap \alpha\colon \mathcal{P}_f S\cdot \{F\colon \overline{S} \mid t.\alpha.s \le \int_F \alpha\})\ , & \end{array}$$

which by Lemma 6.2 is an intersection of Cauchy-closed sets. □

Having shown boundedly-continuous t to yield Cauchy-closed $rp.t$, we turn to the converse: that Cauchy-closed h yields boundedly-continuous $wp.h$. For infinite S it must be done directly, rather than from sublinearity.

Lemma 6.8 *For any relational program h in $\mathcal{H}S$, the corresponding expectation transformer $wp.h$ is boundedly continuous.*

Proof. Let $\mathcal{A}$ be a $\Rrightarrow$-directed and bounded subset of expectations in $\mathcal{P}_B S$; we show that for any $c > 0$ and state s in S

$$wp.h.(\sqcup \mathcal{A}).s \quad \le \quad (\sqcup \alpha\colon \mathcal{A}\cdot wp.h.\alpha.s) + c\ ,$$

which is sufficient for continuity since c may be arbitrarily small. (The other direction is given by monotonicity.)

Define $r\colon = (\sqcup \alpha\colon \mathcal{A}\cdot wp.h.\alpha.s)$; then we have

$$\begin{array}{lll} & (\forall \alpha\colon \mathcal{A}\cdot wp.h.\alpha.s \le r) & \text{definition of } r \\ \text{iff} & (\forall \alpha\colon \mathcal{A}\cdot (\sqcap F\colon h.s\cdot \int_F \alpha) \le r) & \text{Definition 2.6} \\ \text{implies} & (\forall \alpha\colon \mathcal{A}\cdot (\exists F\colon h.s\cdot \int_F \alpha \le r + c)) & c > 0 \\ \text{iff} & (\forall \alpha\colon \mathcal{A}\cdot \{F\colon h.s\cdot \int_F \alpha \le r + c\} \ne \emptyset) & \end{array}$$

$$\begin{array}{lll} \text{implies} & & \text{Lemma 6.3; } h.s \text{ closed in } \overline{S}\text{; } \mathcal{A} \text{ directed; } \overline{S} \text{ compact – see below} \\ & (\cap \alpha\colon \mathcal{A}\cdot \{F\colon h.s\cdot \int_F \alpha \le r + c\}) \ne \emptyset & \\ \text{iff} & (\exists F\colon h.s\cdot (\forall \alpha\colon \mathcal{A}\cdot \int_F \alpha \le r + c)) & \\ \text{iff} & (\exists F\colon h.s\cdot \int_F (\sqcup \mathcal{A}) \le r + c) & \text{bounded monotone convergence} \\ \text{implies} & (\sqcap F\colon h.s\cdot \int_F (\sqcup \mathcal{A})) \le r + c & \\ \text{iff} & wp.h.(\sqcup \mathcal{A}).s \le r + c\ . & \text{Definition 2.6} \end{array}$$

For the deferred justification note that Lemma 6.3 and Cauchy-closure of $h.s$ imply Cauchy-closure of each set $\{F\colon h.s\cdot \int_F \alpha \le r + c\}$, and $\mathcal{A}$'s being directed ensures that they have the finite intersection property. □

Figure 7 illustrates continuity for a transformer over an infinite state space.

var $s\colon \mathbb{N}$;
$s := 0$;
do
  break $\frac{1}{2}\oplus\, s := s+1$
od

This deterministic program takes all initial states to the same final distribution, namely one which assigns probability $1/2^{s+1}$ to state s.
Although the program is 'infinitely branching' – every natural number is reachable with non-zero probability – it is still $\sqcup$-continuous since the branching is probabilistic, not demonic or angelic. (Recall that in contrast infinitely $\sqcap$-branching programs are not generally $\sqcup$-continuous.)

Fig. 7. Continuous program over infinite state space

6.3 The Galois connection

Now we re-establish the partial Galois connection, and with it our characterisation of demonic programs, for infinite state spaces.

Let $\mathcal{T}_C S$ be the boundedly continuous expectation transformers in $\mathcal{T}S$. The fact that rp and *wp* form a partial Galois connection between $\mathcal{H}S$ and $\mathcal{T}_C S$ is not difficult to show: the inequalities

$$rp \circ \mathit{wp} \sqsubseteq id \quad \text{and} \quad id \sqsubseteq \mathit{wp} \circ rp$$

do not require finiteness of S. However our original proofs of the stronger results, namely the equality $rp \circ \mathit{wp} = id$ and that sublinearity characterises the *wp* images in $\mathcal{T}S$ (reported above as Theorem 2.9 [17, Lemm. 8.2, 8.6]), did use finiteness.

Our new proof for the first is as follows.

Lemma 6.9 *For any h in $\mathcal{H}S$ we have*

$$rp.(\mathit{wp}.h) \quad = \quad h\,.$$

Proof. The proof for the finite case [17, Lem. 8.2] appealed to the *separating hyperplane lemma*[18] [17, Lem. A.1]. Using Lemma A.1 (of this paper) instead allows the proof to go through for the infinite case. □

To recover the second result we must strengthen its assumptions to include bounded continuity; then we have

[18] The separating hyperplane lemma is a standard result from linear programming and (specialised to this context) says that given any convex set $\mathcal{C}$ of distributions over a finite state space, and any distribution $F \notin \mathcal{C}$, there is a hyperplane separating F from $\mathcal{C}$.

Lemma 6.10 *If (boundedly-continuous) t in $\mathcal{T}_C S$ is sublinear, then*

$$wp.(rp.t) \quad = \quad t\ .$$

Proof. We are able to replay our earlier proof [17, Lem. 8.6] that established $wp.(rp.t).\alpha \equiv t.\alpha$, but do so only for expectations α with finite support. (Full details are given elsewhere [10].)

Bounded continuity of t then allows the result to be extended to all α: on the left we have that $rp.t$ is Cauchy-closed because t is boundedly continuous, and that $wp.(rp.t)$ is boundedly continuous because $rp.t$ is Cauchy-closed; on the right t itself is boundedly continuous. We therefore take limits on both sides over finitary restrictions $\alpha{\downarrow}T$ of α. □

6.4 Healthiness conditions

The results above show that in our characterisation of *wp*-images of programs in $\mathcal{H}S$, the healthiness conditions Fig. 8 are applicable to the infinite case as well. The first three conditions in Fig. 8 are consequences of sublinearity as shown by our earlier work [17] and, in the special case where the scalars are $\{0,1\}$-valued, all are generalisations of properties of predicate transformers. Sublinearity is the appropriate generalisation of conjunctivity [17, following Lem. 7.2], and feasibility, for example, generalises strictness (another of Dijkstra's original axioms). To see that, we reason first

$$t.\underline{0} \Rrightarrow \sqcup\underline{0} \equiv 0\ ,$$

and then, noting the inequality $t.\underline{0} \Lleftarrow \underline{0}$, we can deduce that in fact $t.\underline{0} \equiv \underline{0}$.

The results of Sections 3–5 also remain valid within the space of boundedly-continuous expectation transformers. All our constructions preserve continuity – however Definition 5.4 must be applied only to finitary expectations, and then Theorem 5.8 establishes any boundedly-continuous and semi-sublinear transformer as a supremum of sublinear ones. As summarised in this section, the details of our proofs where we must appeal to theorems from the theory of convex sets are reduced to reasoning over a finite projection of the state space, where those theorems are still valid.

7 Conclusion

This paper has contributed to the theory of probabilistic programs in two ways: we have explored the structure of the expectation-transformer space (Fig. 9); and we have extended the earlier theory [17], together with the results of the exploration here, to infinite state spaces.

feasibility	$t.\alpha \Rrightarrow \sqcup\underline{\alpha}$	for $\alpha\colon \mathcal{P}_B S$
monotonicity	$\alpha_1 \Lleftarrow \alpha_2$ implies $t.\alpha_1 \Lleftarrow t.\alpha_2$	for $\alpha_1, \alpha_2\colon \mathcal{P}_B S$
scaling	$t.(c\alpha) \equiv c(t.\alpha)$	for $\alpha\colon \mathcal{P}_B S$ and $c\colon \mathbb{R}_{\geq}$
sublinearity	$c_1(t.\alpha_1) + c_2(t.\alpha_2) \ominus \underline{c_0}$ $\Rrightarrow t.(c_1\alpha_1 + c_2\alpha_2 \ominus \underline{c_0})$	for $\alpha_1, \alpha_2\colon \mathcal{P}_B S$ and $c_0, c_1, c_2\colon \mathbb{R}_{\geq}$
bounded continuity	$t.(\sqcup\mathcal{A}) \equiv (\sqcup\alpha\colon \mathcal{A} \cdot t.\alpha)$	for a bounded, $\Rrightarrow$-directed subset $\mathcal{A}$ of $\mathcal{P}_B S$

Fig. 8. The properties characterising $\mathcal{T}_{\sqcap}\, S$ in the infinite case (carried over intact from the finite [17])

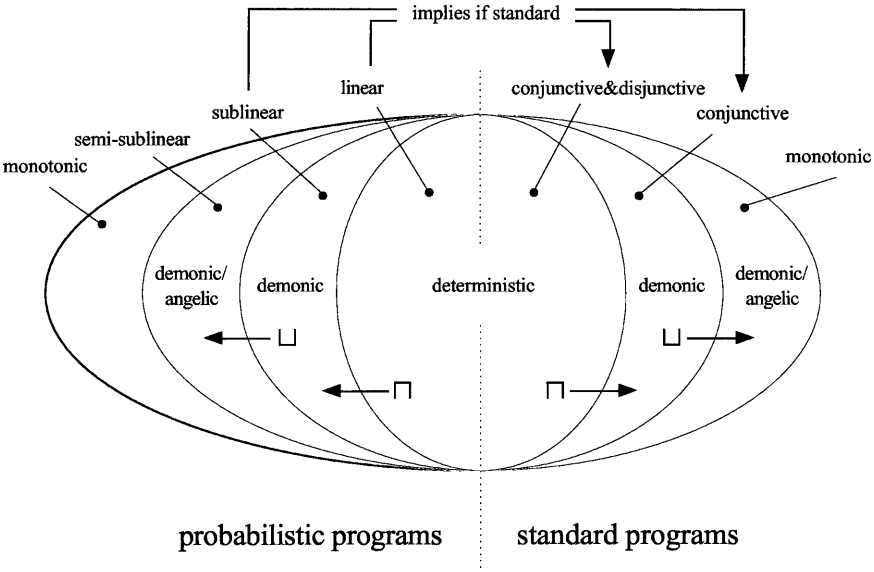

Fig. 9. Structure of transformer spaces. On the standard side, the angelic/demonic programs exhaust the space of monotonic transformers. There are however monotonic probabilistic transformers lying outside of $\mathcal{T}_{\boxplus}\, S$

Even with infinite state spaces, we need only use discrete probability distributions. That is because under the assumption [17] that any implementation of a probabilistic specification can only use the discrete choice ${}_p\oplus$ it follows that all implementations will result only in discrete distributions. An interesting topic would be to reinstate continuous distributions [5] into our programs and then to investigate their effects on the healthiness conditions.

In the standard case, the Back and von Wright's decomposition [1] of the transformer space into deterministic, demonic and angelic components explains the complete behaviour of all monotonic predicate transformers; thus in that theory monotonicity remains as the only surviving restriction

– and a necessary one – since in the activity of program specification and development monotonicity is paramount.

For expectation transformers the situation is very different – our Theorem 5.8 implies that there is still part of the space as yet uninvestigated, namely the monotonic transformers that fail to be semi-sublinear. One such expectation transformer is the *weakest liberal* expectation of a program: applied to a standard post-condition it returns the probability that either the post-condition is established, or that the program never terminates – though monotonic, it is not semi-sublinear. (The definitions [9] and applications [12] of weakest-liberal expectations are given elsewhere.) However the general importance of this part of the space in a programming context is far from clear and it remains a topic for further investigation. Other examples of 'exotic' expectation transformers are the temporal operators [16, 15] that generalise Kozen's predicate transformer definitions [8].

The practical import of our results is to give laws in the (arithmetic) programming logic that are satisfied by programs in the various levels of the hierarchy. Such laws are the basis for derived program-development rules such as the use of variants and invariants for loops [12] and, more generally, for programming algebras and even probabilistic temporal logics [16, 15].

Appendix A: More linear programming lemmas

Lemma A.1 *Let $\mathcal{C}$ be a convex subset of $\mathbb{R}^S$ that is compact (hence closed) in the product $\mathcal{E}_S$ of the Euclidean topologies over its constituent projections $\mathbb{R}$. If some p does not lie in $\mathcal{C}$, then there is a separating hyperplane with p on one side of it and all of $\mathcal{C}$ on the other.*

Proof. If $p \notin \mathcal{C}$, then because $\mathcal{C}$ is closed there is some neighbourhood N in the basis of $\mathcal{E}_S$ with $p \in N$ and $N \cap \mathcal{C} = \emptyset$.

Let T in $\mathbb{F}S$ be the support of N. Writing $(\downarrow T)$ for projection onto T, we then have $p{\downarrow}T \in N{\downarrow}T$ and $N{\downarrow}T \cap \mathcal{C}{\downarrow}T = \emptyset$, because for the latter $N{\downarrow}(S - T) = \mathbb{R}^{S-T}$, and thus $p{\downarrow}T \notin \mathcal{C}{\downarrow}T$. Note that $\mathcal{C}{\downarrow}T$ is compact because $\mathcal{C}$ is, hence closed; and it is convex also because $\mathcal{C}$ is.

Applying the standard separating hyperplane lemma for the finite dimensional space [17, Lem. A1] in the case of $p{\downarrow}T$ and $\mathcal{C}{\downarrow}T$, within the finite-dimensional $\mathbb{R}^T$, gives us a separating hyperplane there; its extension parallel to the remaining axes $S - T$ is then the hyperplane we seek in $\mathbb{R}^S$. □

References

1. R.-J. R. Back, J. von Wright: Duality in specification languages: a lattice theoretical approach. Acta Inf. 27: 583–625 (1990)

2. E.W. Dijkstra: A Discipline of Programming. Englewood Cliffs, N.J.: Prentice Hall 1976
3. G. Grimmett, D. Welsh: Probability: an Introduction. Oxford Science Publications 1986
4. Wim H Hesselink: Programs, Recursion and Unbounded Choice. Number 27 in Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, Cambridge, U.K. 1992
5. C. Jones: Probabilistic nondeterminism. Monograph ECS-LFCS-90-105, Edinburgh Univ. Edinburgh, U.K., 1990. (PhD Thesis)
6. D. Kozen: Semantics of probabilistic programs. J. Comput. Syst. Sci. 22: 328–350 (1981)
7. D. Kozen: A probabilistic PDL. In: Proceedings of the 15th ACM Symposium on Theory of Computing, New York: ACM 1983
8. D. Kozen: Results on the propositional μ-calculus. Theor. Comput. Sci. 27: 333–354 (1983)
9. A. K. McIver, C.C. Morgan: Partial correctness for probabilistic programs. Submitted to Theoretical Computing Science
10. A. McIver, C. Morgan: Probabilistic predicate transformers: part 2. Technical Report PRG-TR-5-96, Programming Research Group, March 1996
11. C. C. Morgan. The specification statement. ACM Trans. Programm. Lang. Syst. 10(3), July 1988
12. C. C. Morgan: Proof rules for probabilistic loops. In: H. Jifeng, J. Cooke, P. Wallis (eds) Proceedings of the BCS-FACS 7th Refinement Workshop, Workshops in Computing. Berlin Heidelberg New York: Springer 1996
13. C. C. Morgan, T. N. Vickers (eds): On the Refinement Calculus. FACIT Series in Computer Science. Berlin Heidelberg New York: Springer 1994
14. C. Morgan: pGCL: Formal reasoning for random algorithms. Proceedings of WOFACS '98, Special Issue of South African Computer Journal Vol 22, March 1999
15. C. Morgan, A. McIver: An expectation-based model for probabilistic temporal logic. Technical Report PRG-TR-20-97, Programming Research Group, 1997
16. C. Morgan, A. McIver: A probabilistic temporal calculus based on expectations. In: L. Groves, S. Reeves (eds) Proc. Formal Methods Pacific '97. Singapore: Springer 1997
17. C. Morgan, A. McIver, K. Seidel: Probabilistic predicate transformers. ACM Trans. Programm. Lang. Syst. 18(3): 325–353 (1996)
18. G. Nelson: A generalization of Dijkstra's calculus. ACM Trans. Programm. Lang. Syst. 11(4): 517–561 (1989)
19. PSG: Probabilistic Systems Group: Collected reports. http://www.comlab.ox.ac.uk/oucl/groups/probs/bibliography.html
20. J. R. Rao: Reasoning about probabilistic parallel programs. ACM Trans. Programm. Lang. Syst. 16(3): 798–842 (1994)