# Elementary probability theory in the Eindhoven style

Carroll Morgan

University of New South Wales, NSW 2052 Australia
carrollm@cse.unsw.edu.au *

**Abstract.** We extend the Eindhoven quantifier notation to elementary probability theory by adding "distribution comprehensions" to it.

Even elementary theories can be used in complicated ways, and this occurs especially when reasoning about computer programs: an instance of this is the multi-level probabilistic structures that arise in probabilistic semantics for security.

Our exemplary case study in this article is therefore the probabilistic reasoning associated with a quantitative noninterference semantics based on Hidden Markov Models of computation. But we believe the proposal here will be more generally applicable than that, and so we also revisit a number of popular puzzles, to illustrate the new notation's wider utility.

## 1 Context and motivation

Conventional notations for elementary probability theory are more descriptive than calculational. They communicate ideas, but they are not algebraic (as a rule) in the sense of helping to proceed reliably from one idea to the next one: and truly effective notations are those that we can reason *with* rather than simply *about*. In our recent work on security, the conventional notations for probability became so burdensome that we felt that it was worth investigating alternative, more systematic notations for their own sake.

The Eindhoven notation was designed in the 1970's to control complexity in reasoning about programs and their associated logics: the forty years since then have shown how effective it is. But as far as we know it has not been used for probability. We have done so by working "backwards," from an application in computer security (Sec. 9.2), with the Eindhoven style as a target (Sec. 2). That is the opposite, incidentally, of reconstructing elementary probability "forwards" from first principles — also a worthwhile goal, but a different one.

We judge our proposal's success by whether it simplifies reasoning about intricate probabilistic structures in computer science and elsewhere. For that we give a small case study, based on noninterference-security semantics, both in the novel notation and in the conventional notation; and we compare them with each other (Sec. 9). We have also used the new notation more extensively [15].

* We are grateful for the support of the Dutch NWO (Grant 040.11.303) and the Australian ARC (Grant DP1092464).

Although the notation was developed retroactively, the account we give here is forwards, that is from the basics towards more advanced constructions. Along the way we use a number of popular puzzles as more general examples.

## 2 The Eindhoven quantifier notation, and our extension

In the 1970's, researchers at *THE* in Eindhoven led by EW Dijkstra proposed a uniform notation for quantifications in first-order logic, elementary set theory and related areas [3]. By $(\mathcal{Q}\, x{:}\, T \mid \mathit{rng} \bullet \mathit{exp})$ [1] they meant that *quantifier* $\mathcal{Q}$ *binds variable* $x$ *of type* $T$ *within textual scope* $(\cdots)$*, that* $x$ *is constrained to satisfy formula rng, that expression exp is evaluated for each such* $x$ *and that those values then are combined via an associative and commutative operator related to quantifier* $\mathcal{Q}$. These examples make the uniformity evident:

| | | |
|---|---|---|
| $(\forall x{:}\, T \mid \mathit{rng} \bullet \mathit{exp})$ | means | for all $x$ in $T$ satisfying *rng* we have *exp*, |
| $(\exists x{:}\, T \mid \mathit{rng} \bullet \mathit{exp})$ | means | for some $x$ in $T$ satisfying *rng* we have *exp* |
| $(\sum x{:}\, T \mid \mathit{rng} \bullet \mathit{exp})$ | means | the sum of all *exp* for $x$ in $T$ satisfying *rng*, |
| $\{x{:}\, T \mid \mathit{rng} \bullet \mathit{exp}\}$ | means | the set of all *exp* for $x$ in $T$ satisfying *rng*. |

A general shorthand applying to them all is that an omitted range *rng* defaults to *true*, and an omitted *exp* defaults to the bound variable $x$ itself.

These (once) novel notations are not very different from the conventional ones: they contain the same ingredients because they must. Mainly they are a reordering, an imposition of consistency, and finally a making explicit of what is often implicit: bound variables, and their scope. Instead of writing $\{n{\in}\mathbb{N} \mid n{>}0\}$ for the positive natural numbers we write $\{n{:}\,\mathbb{N} \mid n{>}0\}$, omitting the "$\bullet n$" via the shorthand above; the only difference is the explicit declaration of $n$ via a colon (as in a programming language) rather than via $n{\in}\mathbb{N}$ which, properly speaking, is a formula (with both $n$ and $\mathbb{N}$ free) and doesn't declare anything. And instead of $\{n^2 \mid n{\in}\mathbb{N}\}$ for the square numbers, we write $\{n{:}\,\mathbb{N} \bullet n^2\}$, keeping the declaration in first position (always) and avoiding ambiguous use of the vertical bar.

In program semantics one can find general structures such as

*sets of distributions* for probability and nondeterminism [21, 14],

*distributions of distributions*, for probabilistic noninterference security [15, 16], and even

*sets of distributions of distributions* to combine the two [17].

All of these are impeded by the conventional use of "Pr" to refer to probability with respect to some unnamed distribution "of interest" at the time: we need to refer to the whole distribution itself.

And when we turn to particular instances, the semantics of individual programs, we need to build functions corresponding to specific program components. The conventional "random variables" are inconvenient for this, since we

[1] The original Eindhoven style uses colons as separators; the syntax here with | and • is one of many subsequent variations based on their innovation.

must invent a name for every single one: we would rather use the expressions and variables occurring in the programs themselves. In the small –but nontrivial– example of information flow (Sec. 9), borrowed from our probabilistic security work [15–17], we compare the novel notation (Sec. 9.2) to the conventional (Sec. 9.3) in those respects.

Our essential extension of the Eindhoven quantifiers was to postulate a "distribution comprehension" notation $\{\!|s{:}\,\delta \mid rng \bullet exp|\!\}$, intending it to mean "for all elements $s$ in the distribution $\delta$, conditioned by $rng$, make a new distribution based on evaluating $exp$." Thus we refer to a distribution itself (the whole comprehension), and we access random variables as expressions (the $exp$ within). From there we worked backwards, towards primitives, to arrange that indeed the comprehension would have that meaning.

This report presents our results, but working forwards and giving simple examples as we go. Only at Def. 13 do we finally recover our conceptual starting point, a definition of the comprehension that agrees with the guesswork just above (Sec. 8.5).

## 3 Discrete distributions as enumerations

We begin with distributions written out explicitly: this is by analogy with the enumerations of sets which list their elements. The notation $f.x$ for application of function $f$ to argument $x$ is used from here on, except for type constructors where a distinct font allows us to reduce clutter by omitting the dot.

### 3.1 Finite discrete distributions as a type

A finite discrete distribution $\delta$ on a set $S$ is a function assigning to each element $s$ in $S$ a (non-negative) probability $\delta.s$, where the sum of all such probabilities on $S$ is one. The fair distribution on coin-flip outcomes $\{\mathsf{H}, \mathsf{T}\}$ takes both $\mathsf{H}, \mathsf{T}$ to $1/2$; the distribution on die-roll outcomes $\{1{\cdot\cdot}6\}$ for a fair die gives $1/6$ for each integer $n$ with $1{\leq}n{\leq}6$. In general we have

**Definition 1.** *The constructor $\mathbb{D}$ for finite discrete distributions*

The set $\mathbb{D}S$ of discrete distributions over a finite set $S$ is the functions from $S$ into $[0,1]$ that sum to one, that is $\{\delta{:}\,S{\to}[0,1] \mid (\sum s{:}\,S \bullet \delta.s) = 1\}$. The set $S$ is called the *base* (type) of $\delta$. □

In Def. 1 the omitted $|rng$ of $\sum$ is $|true$, and the omitted $\bullet exp$ of $\{\cdots\}$ is $\bullet\delta$. One reason for using distinct symbols | and • is that in the default cases those symbols can be omitted as well, with no risk of ambiguity.

### 3.2 The support of a distribution

The support of a distribution is that subset of its base to each of whose elements it assigns nonzero probability; it is in an informal sense the "relevant" or "interesting" elements in the distribution. We define

**Definition 2.** *Support of a distribution* For distribution $\delta: \mathbb{D}S$ with base $S$, the support is the subset $\lceil\delta\rceil := \{s: S \mid \delta.s \neq 0\}$ of $S$. □

The "ceiling" notation $\lceil\cdot\rceil$ suggests the pointwise ceiling of a distribution which, as a function (Def. 1), is the characteristic function of its support.

### 3.3 Specialised notation for uniform distributions

By analogy with set enumerations like $\{\mathsf{H}, \mathsf{T}\}$, we define uniform-distribution enumerations that assign the same probability to every element in their support:

**Definition 3.** *Uniform-distribution enumeration* The uniform distribution over an enumerated set $\{a, b, \cdots, z\}$ is written $\{\!\{a, b, \cdots, z\}\!\}$. □

Thus for example the fair-coin distribution is $\{\!\{\mathsf{H}, \mathsf{T}\}\!\}$ and the fair-die distribution is $\{\!\{1\,..\,6\}\!\}$. (The empty $\{\!\{\}\!\}$ would be a sub-distribution [10, 21], not treated here.)

As a special case of uniform distribution we have the point distribution $\{\!\{a\}\!\}$ on some element $a$, assigning probability 1 to it: this is analogous to the singleton set $\{a\}$ that contains only $a$.

### 3.4 General notation for distribution enumerations

For distributions that are not uniform, we attach a probability explicitly to each element. Thus we have $\{\!\{\mathsf{H}^{@\frac{2}{3}}, \mathsf{T}^{@\frac{1}{3}}\}\!\}$ for the coin that is twice as likely to give heads $\mathsf{H}$ as tails $\mathsf{T}$, and $\{\!\{1^{@\frac{2}{9}}, 2^{@\frac{1}{9}}, 3^{@\frac{2}{9}}, 4^{@\frac{1}{9}}, 5^{@\frac{2}{9}}, 6^{@\frac{1}{9}}\}\!\}$ for the die that is twice as likely to roll odd as even (but is uniform otherwise). In general we have

**Definition 4.** *Distribution enumeration* We write $\{\!\{a^{@p_a}, b^{@p_b}, \cdots, z^{@p_z}\}\!\}$ for the distribution over set $\{a, b, \cdots, z\}$ that assigns probability $p_a$ to element $a$ etc. For well-formedness we require that $p_a + p_b + \cdots + p_z = 1$. □

### 3.5 The support of a distribution is a subset of its base

Strictly speaking one can't tell, just by drawing samples, whether $\{\!\{\mathsf{H}, \mathsf{T}\}\!\}$ represents the distribution of a fair two-sided coin, or instead represents the distribution of a three-sided coin with outcomes $\{\mathsf{H}, \mathsf{T}, \mathsf{E}\}$ that never lands on its edge $\mathsf{E}$. Similarly we might not know whether $\{\!\{6\}\!\}$ describes a die that has the numeral 6 written on every face or a loaded die that always rolls 6.

Saying that $\delta$ is uniform *over* $S$ means it is uniform and its support is $S$.

### 3.6 Specialised infix notations for making distributions

For distributions of support no more than two we have the special notation

**Definition 5.** *Doubleton distribution* For any elements $a, b$ and $0 \leq p \leq 1$ we write $a\ {}_p\!\oplus\ b$ for the distribution $\{\!\{a^{@p}, b^{@1-p}\}\!\}$. □

Thus the fair-coin distribution $\{\!|\mathsf{H},\mathsf{T}|\!\}$ can be written $\mathsf{H}\,{}_{1/2}\!\oplus\mathsf{T}$. For the weighted sum of two distributions we have

**Definition 6.** *Weighted sum* For two numbers $x, y$ and $0{\le}p{\le}1$ we define $x\,{}_p\!+ y{:=}px + (1{-}p)y$; more generally $x, y$ can be elements of a vector space.

In particular, for two distributions $\delta, \delta'{:}\,\mathbb{D}S$ we define their weighted sum $\delta\,{}_p\!+\delta'$ by $(\delta\,{}_p\!+\delta').s := p(\delta.s) + (1{-}p)(\delta'.s)$ for all $s$ in $S$. □

Thus the biased die from Sec. 3.4 can be written as $\{\!|1,3,5|\!\}\,{}_{2/3}\!+\{\!|2,4,6|\!\}$, showing at a glance that its odds and evens are uniform on their own, but that collectively the odds are twice as likely as the evens.

As simple examples of algebra we have first $x\,{}_p\!\oplus y = \{\!|x|\!\}\,{}_p\!+\{\!|y|\!\}$, and then

$$\delta\,{}_0\!+\delta' = \delta' \qquad \text{and} \quad \delta\,{}_1\!+\delta' = \delta$$
$$\text{and} \quad \lceil\delta\,{}_p\!+\delta'\rceil = \lceil\delta\rceil \cup \lceil\delta'\rceil \text{ when } 0{<}p{<}1.$$

### 3.7 Comparison with conventional notation [2]

Conventionally a distribution is over a *sample space* $S$, which we have called the base (Def. 1). Subsets of the sample space are *events*, and a distribution assigns a number to every event, the probability that an observation "sampled" from the sample space will be an occurrence of that event. That is, a distribution is of type $\mathbb{P}S{\to}[0,1]$ from subsets of $S$ rather than from its elements.

With our odd-biased die in Sec. 3.4 the sample space is $S{=}\{1{\cdot\cdot}6\}$ and the probability 2/3 of "rolled odd," that is of the event $\{1,3,5\}{\subset}S$, is twice the probability 1/3 of "rolled even," that is of the event $\{2,4,6\}{\subset}S$.

There are "additivity" conditions placed on general distributions, among which are that the probability assigned to the union of two disjoint events should be the sum of the probabilities assigned to the events separately, that the probability assigned to all of $S$ should be one, and that the probability assigned to the empty event should be zero.

When $S$ is finite, the general approach specialises so that a *discrete* distribution $\delta$ acts on separate points, instead of on sets of them. The probability of any event $S'{\subset}S$ is then just $\sum_{s\in S'}\delta(s)$ from additivity.

## 4 Expected values over discrete distributions

### 4.1 Definition of expected value as average

If the base $S$ of a distribution $\delta{:}\,\mathbb{D}S$ comprises numbers or, more generally, is a vector space, then the "weighted average" of the distribution is the sum of the values in $S$ multiplied by the probability that $\delta$ assigns to each, that is $(\sum s{:}\,S \bullet \delta.s{\times}s)$. For the fair die that becomes $(1{+}2{+}3{+}4{+}5{+}6)/6 = 3\frac{1}{2}$; for the odd-biased die the average is $4\frac{2}{3}$.

For the fair coin $\{\!|\mathsf{H},\mathsf{T}|\!\}$ however we have no average, since $\{\mathsf{H},\mathsf{T}\}$ has no arithmetic. We must work indirectly via a function on the base, using

[2] In these comparison sections we will use conventional notation throughout, for example writing $f(x)$ instead of $f.x$ and $\{exp \mid x{\in}S\}$ instead of $\{x{:}\,S \bullet exp\}$.

**Definition 7.** *Expected value* By $(\mathcal{E}s{:}\,\delta \bullet exp)$ we mean the expected value of function $(\boldsymbol{\lambda}\, s \bullet exp)$ over distribution $\delta$; it is

$$(\mathcal{E}s{:}\,\delta \bullet exp) \quad := \quad (\textstyle\sum s{:}\,\lceil\delta\rceil \bullet \delta.s \times exp)\ . \quad [3]$$

Note that *exp* is an expression in which bound variable $s$ probably appears (though it need not). We call *exp* the *constructor*. □

For example, the expected value of the *square* of the value rolled on a fair die is $(\mathcal{E}s{:}\,\{\!|1\,..\,6|\!\} \bullet s^2) = (1^2 + \cdots + 6^2)/6 = 15\frac{1}{6}$.

For further examples, we name a particular distribution $\Delta{:=}\{\!|0,1,2|\!\}$ and describe a notation for converting Booleans to numbers:

**Definition 8.** *Booleans converted to numbers* The function $[\cdot]$ takes Booleans $\mathsf{T}, \mathsf{F}$ to numbers 0,1 so that $[\mathsf{T}]{:=}1$ and $[\mathsf{F}]{:=}0$. [4] □

Then we have

$$\begin{array}{rrcl} & (\mathcal{E}s{:}\,\Delta \bullet s \bmod 2) & = & 1/3{\times}0 + 1/3{\times}1 + 1/3{\times}0 \;=\; 1/3 \\ \text{and} & (\mathcal{E}s{:}\,\Delta \bullet [s{\neq}0]) & = & 1/3{\times}0 + 1/3{\times}1 + 1/3{\times}1 \;=\; 2/3\ , \end{array}$$

where in the second case we have used Def. 8 to convert the Boolean $s{\neq}0$ to a number. Now we can formulate the average proportion of heads shown by a fair coin as $(\mathcal{E}s{:}\,\{\!|\mathsf{H},\mathsf{T}|\!\} \bullet [s{=}\mathsf{H}]) = \frac{1}{2}$.

### 4.2 The probability of a subset rather than of a single element

We can use the expected value quantifier to give the aggregate probability assigned to a (sub)set of outcomes, provided we have a formula describing that set. [5] When *exp* is Boolean, we have that $(\mathcal{E}s{:}\,\delta \bullet [exp])$ is the probability assigned by $\delta$ to the whole of the set $\{s{:}\,\lceil\delta\rceil \mid exp\}$. This is because the expected value of the characteristic function of a set is equal to the probability of that set as a whole. An example of this is given at 4.3(e) below.

### 4.3 Abbreviation conventions

The following are five abbreviations that we use in the sequel.

(a) If several bound variables are drawn from the same distribution, we assume they are drawn independently from separate instances of it. Thus $(\mathcal{E}x, y{:}\,\delta \bullet \cdots)$ means $(\mathcal{E}x{:}\,\delta, y{:}\,\delta \bullet \cdots)$ or equivalently $(\mathcal{E}(x,y){:}\,\delta^2 \bullet \cdots)$.

[3] Here is an example of not needing to know the base type: we simply sum over the support of $\delta$, since the other summands will be zero anyway.

[4] We disambiguate $\mathsf{T}$ for *true* and $\mathsf{T}$ for *tails* by context.

[5] Note that those aggregate probabilities do not sum to one over all subsets of the base, since the individual elements would be counted many times.

(b) If in an expected-value quantification the *exp* is omitted, it is taken to be the bound variable standing alone (or a tuple of them, if there are several). Thus $(\mathcal{E}s{:}\,\delta)$ means $(\mathcal{E}s{:}\,\delta \bullet s)$, and more generally $(\mathcal{E}x, y{:}\,\delta)$ means $(\mathcal{E}x, y{:}\,\delta \bullet (x, y))$ with appropriate arithmetic induced on $\lceil\delta\rceil\times\lceil\delta\rceil$.
(c) By analogy with summation, where for a set $S$ we abbreviate $(\sum s{:}\,S)$ by $\sum S$, we abbreviate $(\mathcal{E}s{:}\,\delta)$ by $\mathcal{E}\delta$. Thus $\mathcal{E}\Delta = \mathcal{E}\{\!|0,1,2|\!\} = (0{+}1{+}2)/3 = 1$.
(d) If a set is written where a distribution is expected, we assume implicitly that it is the uniform distribution over that set. Thus $\mathcal{E}\{0,1,2\} = \mathcal{E}\Delta = 1$.
(e) If a Boolean expression occurs where a number is expected, then we assume an implicit application of the conversion function $[\cdot]$ from Def. 8. Thus $(\mathcal{E}s{:}\,\{0,1,2\} \bullet s{\neq}0) = 2/3$ is the probability that a number chosen uniformly from $0, 1, 2$ will not be zero.

### 4.4 Example of expected value: dice at the fairground

Define the set $D$ to be $\{1{\cdot\cdot}6\}$, the possible outcomes of a die roll.

At the fairground there is a tumbling cage with three fair dice inside, and a grid of six squares marked by numbers from $D$. You place \$1 on a square, and watch the dice tumble until they stop.

If your number appears exactly once among the dice, then you get your \$1 back, plus \$1 more; if it appears twice, you get \$2 more; if it appears thrice you get \$3 more. If it's not there at all, you lose your \$1.

Using our notation so far, your expected profit is written

$$-1 \;+\; (\mathcal{E}s_1, s_2, s_3{:}\,D \bullet (\textstyle\bigvee i \bullet s_i{=}s) + (\sum i \bullet s_i{=}s)) \;, \tag{1}$$

where the initial $-1$ accounts for the dollar you paid to play, and the free variable $s$ is the number of the square on which you placed it. The disjunction describes the event that you get your dollar back; and the summation describes the extra dollars you (might) get as well.

The $D$ is converted to a uniform distribution by 4.3(d), then replicated three times by 4.3(a), independently for $s_{\{1,2,3\}}$; and the missing conversions from Boolean to 0,1 are supplied by 4.3(e).

Finally we abuse notation by writing $s_i$ even though $i$ is itself a (bound) variable: e.g. by $(\bigvee i \bullet s_i{=}s)$ we mean in fact $s_1{=}s \vee s_2{=}s \vee s_3{=}s$. [6]

[6] It is an abuse because in the scope of $i$ we are using it as if it were an argument to some function $s_{(\cdot)}$ — but the name $s$ is already used for something else. Moreover $s_1, s_2, s_3$ must themselves be names(not function applications) since we quantify over them with $\mathcal{E}$. Also we gave no type for $i$.

Although our purpose is to show how we achieve a concise presentation with precise notation, we are at the same time arguing that "to abuse, or not to abuse" should be decided on individual merits. There are times when a bit of flexibility is helpful: arguably the abuse here gains more in readability than it loses in informality.

A similar use is $(\exists i \bullet \cdots H_i \cdots)$ for the weakest precondition of a loop: this finesse avoided swamping a concise first-order presentation with (mostly unnecessary) higher-order logic throughout [2].

While the point of this example is the way in which (1) is written, it's worth pointing out that its value is approximately $-.08$, independent of $s$, thus an expected loss of about eight cents in the dollar every time you play and no matter which square you choose.

### 4.5 Comparison with conventional notation

Conventionally, expected values are taken over *random variables* that are functions from the sample space into a set with arithmetic, usually the reals (but more generally a vector space). Standard usage is first to define the sample space, then to define a distribution over it, and finally to define a random variable over the sample space and give it a name, say $X$. Then one writes $\Pr(X{=}x)$ for the probability assigned by that distribution to the event that the (real-valued) random variable $X$ takes some (real) value $x$; and $\mathbf{E}(X)$ is the notation for the expected value of random variable $X$ over the same (implicit) distribution.

In Def. 7 our random variable is $(\boldsymbol{\lambda}\, s \bullet \mathit{exp})$, and we can write it without a name since its bound variable $s$ is already declared. Furthermore, because we give the distribution $\delta$ explicitly, we can write expressions in which the distributions are themselves expressions. As examples, we have

$$\begin{array}{rll} (\mathcal{E}s{:}\, \{\!|e|\!\} \bullet \mathit{exp}) = \mathit{exp}[s\backslash e] & & \text{– one-point rule} \\ (\mathcal{E}s{:}\, (\delta \;{}_p{+}\; \delta') \bullet \mathit{exp}) = (\mathcal{E}s{:}\, \delta \bullet \mathit{exp}) \;{}_p{+}\; (\mathcal{E}s{:}\, \delta' \bullet \mathit{exp}) & & \text{– using Def. 6} \\ (\mathcal{E}s{:}\, (x \;{}_p{\oplus}\; y) \bullet \mathit{exp}) = \mathit{exp}[s\backslash x] \;{}_p{+}\; \mathit{exp}[s\backslash y] & & \text{– from the two above,} \end{array}$$

where $\mathit{exp}[s\backslash e]$ is bound-variable-respecting replacement of $s$ by $e$ in $\mathit{exp}$.

## 5 Discrete distributions as comprehensions

### 5.1 Definition of distribution comprehensions

With a comprehension, a distribution is defined by properties rather than by enumeration. Just as the set comprehension $\{s{:}\, \lceil \Delta \rceil \bullet s^2\}$ gives the set $\{0, 1, 4\}$ having the property that its elements are precisely the squares of the elements of $\lceil \Delta \rceil = \{0, 1, 2\}$, we would expect $\{\!|s{:}\, \Delta \bullet s^2|\!\}$ to be $\{\!|0, 1, 4|\!\}$ where in this case the uniformity of the source $\Delta$ has induced uniformity in the target.

If however some of the target values "collide," because $\mathit{exp}$ is not injective, then their probabilities add together: thus we have $\{\!|s{:}\, \Delta \bullet s \,\mathbf{mod}\, 2|\!\} = \{\!|0^{@\frac{2}{3}}, 1^{@\frac{1}{3}}|\!\} = 0 \;{}_{2/3}{\oplus}\; 1$, where target element 0 has received $1/3$ probability as $0 \,\mathbf{mod}\, 2$ and another $1/3$ as $2 \,\mathbf{mod}\, 2$.

We define distribution comprehensions by giving the probability they assign to an arbitrary element; thus

**Definition 9.** *Distribution comprehension* For distribution $\delta$ and arbitrary value $e$ of the type of $\mathit{exp}$ we define

$$\{\!|s{:}\, \delta \bullet \mathit{exp}|\!\}.e \quad := \quad (\mathcal{E}s{:}\, \delta \bullet [\mathit{exp}{=}e])\ . \quad {}^{7}$$

□

The construction is indeed a distribution on $\{s{:}\lceil\delta\rceil \bullet \mathit{exp}\}$ (Lem. 1 in App. C), and assigns to element $e$ the probability that $\mathit{exp}{=}e$ as $s$ ranges over $\lceil\delta\rceil$. [8]

### 5.2 Examples of distribution comprehensions

We have from Def. 9 that the probability $\{\!|s{:}\Delta \bullet s \bmod 2|\!\}$ assigns to 0 is

$$\begin{array}{ll} & \{\!|s{:}\Delta \bullet s \bmod 2|\!\}.0 \\ = & (\mathcal{E}s{:}\Delta \bullet [s \bmod 2 = 0]) \\ = & 1/3{\times}[0{=}0] + 1/3{\times}[1{=}0] + 1/3{\times}[0{=}0] \\ = & 1/3{\times}1 + 1/3{\times}0 + 1/3{\times}1 \\ = & 2/3\ , \end{array}$$

and the probability $\{\!|s{:}\Delta \bullet s \bmod 2|\!\}$ assigns to 1 is

$$\begin{array}{ll} & \{\!|s{:}\Delta \bullet s \bmod 2|\!\}.1 \\ = & 1/3{\times}[0{=}1] + 1/3{\times}[1{=}1] + 1/3{\times}[0{=}1] \\ = & 1/3\ . \end{array}$$

Thus we have verified that $\{\!|s{:}\Delta \bullet s \bmod 2|\!\} = 0\ {}_{2/3}\!\oplus 1$ as stated in Sec. 5.1.

### 5.3 Comparison with conventional notation

Conventionally one makes a target distribution from a source distribution by "lifting" some function that takes the source sample space into a target. We explain that here using the more general view of distributions as functions of *subsets* of the sample space (Sec. 3.7), rather than as functions of single elements.

If $\delta_X$ is a distribution over sample space $X$, and we have a function $f{:}X{\to}Y$, then distribution $\delta_Y$ over $Y$ is defined $\delta_Y(Y'){:=}\,\delta_X(f^{-1}(Y'))$ for any subset $Y'$ of $Y$. We then write $\delta_Y = f_*(\delta_X)$, and function $f_*{:}\,\mathbb{D}X{\to}\mathbb{D}Y$ is called the *push-forward*; it makes the *image measure* wrt. $f{:}X{\to}Y$ [5, index].

In the distribution comprehension $\{\!|s{:}\delta \bullet \mathit{exp}|\!\}$ for $\delta{:}\,\mathbb{D}S$, the source distribution is $\delta$ and the function $f$ between the sample spaces is $(\boldsymbol{\lambda}\, s{:}S \bullet \mathit{exp})$. The induced push-forward $f_*$ is then the function $(\boldsymbol{\lambda}\, \delta{:}\,\mathbb{D}S \bullet \{\!|s{:}\delta \bullet \mathit{exp}|\!\})$.

---

[7] Compare $\{x{:}\,X \bullet \mathit{exp}\}\ni e$ defined to be $(\exists x{:}\,X \bullet \mathit{exp}{=}e)$.

[8] A similar comprehension notation is used in cryptography, for example the

$$\{s \xleftarrow{R} S; s' \xleftarrow{R} S' : \mathit{exp}\}$$

that in this case takes bound variables $(s, s')$ uniformly $(\xleftarrow{R})$ from sample spaces $(S, S')$ and, with them, makes a new distribution via a constructor expression $(\mathit{exp})$ containing those variables. We would write that as $\{\!|s{:}\,S; s'{:}\,S' \bullet \mathit{exp}|\!\}$ with the $S, S'$ converted to uniform distributions by 4.3(d).

# 6 Conditional distributions

## 6.1 Definition of conditional distributions

Given a distribution and an event, the latter a subset of possible outcomes, a conditioning of that distribution by the event is a new distribution formed by restricting attention to that event and ignoring all other outcomes. For that we have

**Definition 10.** *Conditional distribution* Given a distribution $\delta$ and a "range" predicate *rng* in variable $s$ ranging over the base of $\delta$, the *conditional distribution of $\delta$ given rng* is determined by

$$\{\!|s\!:\delta \mid rng|\!\}.s' \quad := \quad \frac{(\mathcal{E}s\!:\delta \bullet rng \times [s{=}s'])}{(\mathcal{E}s\!:\delta \bullet rng)} \ ,$$

for any $s'$ in the base of $\delta$. We appeal to the abbreviation 4.3(e) to suppress the explicit conversion $[rng]$ on the right. [9]

The denominator must not be zero (Lem. 2 in App. C). □

In Def. 6.1 the distribution $\delta$ is initially restricted to the subset of the sample space defined by *rng* (in the numerator), potentially making a subdistribution because it no longer sums to one. It it restored to a full distribution by normalisation, the effect of dividing by its weight (the denominator).

## 6.2 Example of conditional distributions

A simple case of conditional distribution is illustrated by the uniform distribution $\Delta = \{\!|0,1,2|\!\}$ we defined earlier. If we condition on the event "is not zero" we find that $\{\!|s\!:\Delta \mid s{\neq}0|\!\} = \{\!|1,2|\!\}$, that when $s$ is not zero it is equally likely to be 1 or 2. We verify this via Def. 10 and the calculation

$$\begin{array}{ll} & \{\!|s\!:\Delta \mid s{\neq}0|\!\}.1 \\ = & (\mathcal{E}s\!:\{\!|0,1,2|\!\} \bullet [s{\neq}0] \times [s{=}1]) \ / \ (\mathcal{E}s\!:\{\!|0,1,2|\!\} \bullet [s{\neq}0]) \\ = & \frac{1}{3}/\frac{2}{3} \\ = & 1/2 \ . \end{array}$$

## 6.3 Comparison with conventional notation

Conventionally one refers to the conditional probability of an event $A$ given some (other) event $B$, writing $\Pr(A|B)$ whose meaning is given by the Bayes formula $\Pr(A{\wedge}B)/\Pr(B)$. Both $A, B$ are names (not expressions) referring to events defined in the surrounding text, and Pr refers, in the usual implicit way, to the probability distribution under consideration. Well-definedness requires that $\Pr(B)$ be nonzero.

Def. 10 with its conversions 4.3(e) explicit becomes

$$(\mathcal{E}s\!:\delta \bullet [s{=}s' \wedge rng]) \ / \ (\mathcal{E}s\!:\delta \bullet [rng]) \ ,$$

with Event $A$ corresponding to "is equal to $s'$" and Event $B$ to "satisfies *rng*."

[9] Leaving the $[\cdot]$ out enables a striking notational economy in Sec. 8.2.

# 7 Conditional expectations

## 7.1 Definition of conditional expectations

We now put constructors *exp* and ranges *rng* together in a single definition of conditional expectation, generalising conditional distributions:

**Definition 11.** *Conditional expectation* Given a distribution $\delta$, predicate *rng* and expression *exp* both in variable $s$ ranging over the base of $\delta$, the *conditional expectation of exp over $\delta$ given rng* is

$$(\mathcal{E}s{:}\,\delta \mid rng \bullet exp) \quad := \quad \frac{(\mathcal{E}s{:}\,\delta \bullet rng \times exp)}{(\mathcal{E}s{:}\,\delta \bullet rng)} \ , \quad {}^{10}$$

in which the expected values on the right are in the simpler form to which Def. 7 applies, and *rng*,*exp* are converted if necessary according to 4.3(e).

The denominator must not be zero. □

## 7.2 Conventions for default range

If *rng* is omitted in $(\mathcal{E}s{:}\,\delta \mid rng \bullet exp)$ then it defaults to $\mathsf{T}$, that is *true* as a Boolean or 1 as a number: and this agrees with Def. 7. To show that, in this section only we use $\underline{\mathcal{E}}$ for Def. 11 and reason

| | | |
|---|---|---|
| | $(\underline{\mathcal{E}}s{:}\,\delta \bullet exp)$ | "as interpreted in Def. 11" |
| = | $(\underline{\mathcal{E}}s{:}\,\delta \mid \mathsf{T} \bullet exp)$ | "default *rng* is $\mathsf{T}$" |
| = | $(\mathcal{E}s{:}\,\delta \bullet [\mathsf{T}] \times exp) \ / \ (\mathcal{E}s{:}\,\delta \bullet [\mathsf{T}])$ | "Def. 11 and 4.3(e)" |
| = | $(\mathcal{E}s{:}\,\delta \bullet exp) \ / \ (\mathcal{E}s{:}\,\delta \bullet 1)$ | "$[\mathsf{T}]$=1" |
| = | $(\mathcal{E}s{:}\,\delta \bullet exp)$ . | "$(\mathcal{E}s{:}\,\delta \bullet 1) = (\sum s{:}\,S \bullet \delta.s) = 1$" |

More generally we observe that a nonzero range *rng* can be omitted whenever it contains no free $s$, of which "being equal to the default value $\mathsf{T}$" is a special case. That is because it can be distributed out through the $(\mathcal{E}s)$ and then cancelled.

## 7.3 Examples of conditional expectations

In our first example we ask for the probability that a value chosen according to distribution $\Delta$ will be less than two, given that it is not zero.

Using the technique of Sec. 4.2 we write $(\mathcal{E}s{:}\,\Delta \mid s{\neq}0 \bullet s{<}2)$ which, via Def. 11, is equal to 1/2. Our earlier example at Sec. 6.2 also gives 1/2, the probability of being less than two in the uniform distribution $\{\!|1, 2|\!\}$.

Our second example is the expected value of a fair die roll, given that the outcome is odd. That is written $(\mathcal{E}s{:}\,D \mid s \bmod 2 = 1)$, using the abbreviation of 4.3(b) to omit the constructor $s$. Via Def. 11 it evaluates to $(1{+}3{+}5)/3 = 3$.

[10] From (9) in Sec. 11 we will see this equivalently as $(\mathcal{E}s{:}\,\{\!|s{:}\,\delta \mid rng|\!\}\bullet exp)$.

### 7.4 Comparison with conventional notation

Conventionally one refers to the expected value of some random variable $X$ given that some other random variable $Y$ has a particular value $y$, writing $\mathbf{E}(X|Y{=}y)$. With $X, Y$ and the distribution referred to by $\mathbf{E}$ having been fixed in the surrounding text, the expression's value is a function of $y$.

Our first example in Sec. 7.3 is more of conditional probability than of conditional expectation: we would state in the surrounding text that our distribution is $\Delta$, that event $A$ is "is nonzero" and event $B$ is "is less than two." Then we would have $\Pr(A|B) = 1/2$.

In our second example, the random variable $X$ is the identity on $D$, the random variable $Y$ is the $\mathbf{mod}\,2$ function, the distribution is uniform on $D$ and the particular value $y$ is 1. Then we have $\mathbf{E}(X|Y{=}1) = 3$.

## 8 Belief revision: *a priori* and *a posteriori* reasoning

### 8.1 A-priori and a-posteriori distributions in conventional style: introduction and first example

*A priori*, i.e. "before" and *a posteriori*, i.e. "after" distributions refer to situations in which a distribution is known (or believed) and then an observation is made that changes one's knowledge (or belief) in retrospect. This is sometimes known as *Bayesian belief revision*. A typical real-life example is the following.

> In a given population the incidence of a disease is believed to be one person in a thousand. There is a test for the disease that is 99% accurate. A patient who arrives at the doctor is therefore *a priori* believed to have only a 1/1,000 chance of having the disease; but then his test returns positive. What is his *a posteriori* belief that he has the disease?

The patient probably thinks the chance is now 99%. But the accepted Bayesian analysis is that one compares the probability of having the disease, and testing positive, with the probability of testing positive on its own (i.e. including false positives). That gives for the *a posteriori* belief

$$\begin{array}{l} \quad \Pr(\text{has disease} \wedge \text{test positive}) \;/\; \Pr(\text{test positive}) \\ = (1/1000) \times (99/100) \;/\; ((1/1000) \times (99/100) + (999/1000) \times (1/100)) \\ = 99 \;/\; (99 + 999) \\ \approx 9\% \;, \end{array}$$

that is less than one chance in ten, and not 99% at all. Although he is believed one hundred times more likely than before to have the disease, still it is ten times less likely than he feared.

### 8.2 Definition of *a posteriori* expectation

We begin with expectation rather than distribution, and define

**Definition 12.** A posteriori *expectation* Given a distribution $\delta$, an experimental outcome *rng* and expression *exp* both possibly containing variable $s$ ranging over the base set of $\delta$, the a posteriori *conditional expectation of exp over $\delta$ given rng* is $(\mathcal{E}s{:}\,\delta \mid rng \bullet exp)$, as in Def. 11 but without requiring *rng* to be Boolean. □

This economical reuse of the earlier definition, hinted at in Sec. 6.1, comes from interpreting *rng* not as a predicate but rather as the probability, depending on $s$, of observing some result. Note that since it varies with $s$ it is not (necessarily) based on any *single* probability distribution, as we now illustrate.

### 8.3 Second example of belief revision: Bertrand's Boxes

Suppose we have three boxes, identical in apprearance and named Box 0, Box 1 and Box 2. Each one has two balls inside: Box 0 has two black balls, Box 1 has one white- and one black ball; and Box 2 has two white balls.

A box is chosen at random, and a ball is drawn randomly from it. *Given that the ball was white*, what is the chance the other ball is white as well?

Using Def. 12 we describe this probability as $(\mathcal{E}b{:}\,\Delta \mid b/2 \bullet b{=}2)$, exploiting the box-numbering convention to write $b/2$ for the probability of observing the event "ball is white" if drawing randomly from Box $b$. Since $(\sum b{:}\,\{0,1,2\} \bullet b/2)$ is $3/2 \neq 1$, it's clear that $b/2$ is not based on some single distribution, even though it is a probability. Direct calculation based on Def. 12 gives

$$\begin{array}{ll} & (\mathcal{E}b{:}\,\Delta \mid b/2 \bullet b{=}2) \\ = & (\mathcal{E}b{:}\,\{\!\{0,1,2\}\!\} \bullet b/2 \times [b{=}2]) \;/\; (\mathcal{E}b{:}\,\{\!\{0,1,2\}\!\} \bullet b/2) \\ = & \frac{1}{3}{\times}\frac{2}{2} \;/\; (\frac{1}{3}{\times}\frac{0}{2} + \frac{1}{3}{\times}\frac{1}{2} + \frac{1}{3}{\times}\frac{2}{2}) \\ = & \frac{1}{3} \;/\; \frac{1}{2} \\ = & 2/3\ . \end{array}$$

The other ball is white with probability 2/3.

### 8.4 Third example of belief revision: The Reign in Spain

In Spain the rule of succession is currently that the next monarch is the eldest son of the current monarch, if there is a son at all: thus an elder daughter is passed over in favour of a younger son. We suppose that the current king had one sibling at the time he succeeded to the throne. What is the probability that his sibling was a brother? [11]

The answer to this puzzle will be of the form

$$(\mathcal{E}\,\mathsf{two\ siblings} \mid \mathsf{one\ is\ king} \bullet \mathsf{the\ other\ is\ male})\ ,$$

[11] We see this as belief revision if we start by assuming the monarch's only sibling is as likely to be male as female; when we learn that the monarch is a Spanish king, we revise our belief.

and we deal with the three phrases one by one.

For two siblings we introduce two Boolean variables $c_{\{0,1\}}$, that is $c$ for "child" and with the larger subscript 1 denoting the child with the larger age (i.e. the older one). Value $\mathsf{T}$ means "is male," and each Boolean will be chosen uniformly, reflecting the an assumption that births are fairly distributed between the two genders.

For the other is male we write $c_0 \wedge c_1$ since the king himself is male, and therefore his sibling is male just when they both are. We have now reached

$$(\mathcal{E}c_0, c_1 \colon \mathsf{Bool} \mid \mathsf{one\ is\ king} \bullet c_0 \wedge c_1) \ . \tag{2}$$

In the Spanish system, there will be a king (as opposed to a queen) just when *either* sibling is male: we conclude our "requirements analysis" with the formula

$$(\mathcal{E}c_0, c_1 \colon \mathsf{Bool} \mid c_0 \vee c_1 \bullet c_0 \wedge c_1) \ . \tag{3}$$

It evaluates to 1/3 via Def. 12: in Spain, kings are more likely to have sisters.

Proceeding step-by-step as we did above allows us easily to investigate alternative situations. What would the answer be in Britain, where the eldest sibling becomes monarch regardless of gender? In that case we would start from (2) but reach the final formulation $(\mathcal{E}c_0, c_1 \colon \mathsf{Bool} \mid c_1 \bullet c_0 \wedge c_1)$ instead of the Spanish formulation (3) we had before. We could evaluate this directly from Def. 12; but more interesting is to illustrate the algebraic possibilities for simplifying it:

$$\begin{array}{lll}
 & (\mathcal{E}c_0, c_1 \colon \mathsf{Bool} \mid c_1 \bullet c_0 \wedge c_1) & \text{"British succession"} \\
= & (\mathcal{E}c_0, c_1 \colon \mathsf{Bool} \mid c_1 \bullet c_0 \wedge \mathsf{T}) & \text{"}c_1 \text{ is } \mathsf{T}\text{, from the range"} \\
= & (\mathcal{E}c_0, c_1 \colon \mathsf{Bool} \mid c_1 \bullet c_0) & \text{"Boolean identity"} \\
= & (\mathcal{E}c_0 \colon \mathsf{Bool} \mid (\mathcal{E}c_1 \colon \mathsf{Bool} \bullet c_1) \bullet c_0) & \text{"}c_1 \text{ not free in constructor } (\bullet c_0)\text{: see below"} \\
= & (\mathcal{E}c_0 \colon \mathsf{Bool} \mid 1/2 \bullet c_0) & \text{"Def. 7"} \\
= & (\mathcal{E}c_0 \colon \mathsf{Bool} \bullet c_0) & \text{"remove constant range: recall Sec. 7.2"} \\
= & 1/2 \ . & \text{"Def. 7"}
\end{array}$$

We set the above out in unusually small steps simply in order to illustrate its (intentional) similarity with normal quantifier-based calculations. The only non-trivial step was the one labelled "see below": it is by analogy with the set equality $\{s \colon S; s' \colon S' \mid rng \bullet exp\} = \{s \colon S \mid (\exists s' \colon S' \bullet rng) \bullet exp\}$ that applies when $s'$ is not free in $exp$. We return to it in Sec. 11.

### 8.5 General distribution comprehensions

Comparison of Def. 10 and Def. 12 suggests a general form for distribution comprehensions, comprising both a range and a constructor. It is

**Definition 13.** *General distribution comprehensions* Given a distribution $\delta$, an experimental outcome $rng$ in variable $s$ that ranges over the base set of $\delta$ and a constructor $exp$, the general *a posteriori* distribution formed via that constructor is determined by

$$\{\!|s \colon \delta \mid rng \bullet exp|\!\}.e \quad := \quad (\mathcal{E}s \colon \delta \mid rng \bullet [exp{=}e]) \ ,$$

for arbitrary $e$ of the type of $exp$. □

Thus $\{\!\{c_0, c_1 : \mathsf{Bool} \mid c_0 \vee c_1 \bullet c_0 \wedge c_1\}\!\} = \mathsf{T}_{1/3} \oplus \mathsf{F}$, giving the distribution of kings' siblings in Spain.

### 8.6 Comparison with conventional notation

Conventional notation for belief revision is similar to the conventional notation for conditional reasoning once we take the step of introducing the *joint distribution*.

In the first example, from Sec. 8.1, we would consider the joint distribution over the product space, that is

| Joint sample space (Cartesian product) | Joint distribution $\times 100,000$ | |
|---|---|---|
| | ⊞ | ⊟ |
| {has disease ☹, doesn't have disease ☺} | ☹ 1×99 | 1×1 |
| × {test positive ⊞, test negative ⊟} | ☺ 999×1 | 999×99 |

and then the column corresponding to ⊞, i.e. test positive, assigns weights 99 and 999 to ☹ and ☺ respectively. Normalising those weights gives the distribution $☹ \,{}_{9\%}\!\oplus ☺$ for the *a posteriori* health of the patient.

Thus we would establish that joint distribution, in the surrounding text, as the referent of Pr, then define as random variables the two projection functions $H$ (health) and $T$ (test), and finally write for example $\Pr(H{=}☹ | T{=}⊞) = 9\%$ for the *a posteriori* probability that a positive-testing patient has the disease.

## 9 Use in computer science for program semantics

### 9.1 "Elementary" can still be intricate

By *elementary* probability theory we mean discrete distributions, usually over finite sets. Non-elementary would then include measures, and the subtle issues of measurability as they apply to infinite sets. In Sec. 9.2 we illustrate how simple computer programs can require intricate probabilistic reasoning even when restricted to discrete distributions on small finite sets.

The same intricate-though-elementary effect led to the Eindhoven notation in the first place.

A particular example is assignment statements, which are mathematically elementary: functions from state to state. Yet for *specific* program texts those functions are determined by expressions in the program variables, and they leave most of those variables unchanged: working with syntactic substitutions is a better approach [2, 3], but that can lead to complex formulae in the program logic.

Careful control of variable binding, and quantifiers, reduces the risk of reasoning errors in the logic, and can lead to striking simplifications because of the algebra that a systematic notation induces. That is what we illustrate in the following probabilistic example.

### 9.2 Case study: quantitative noninterference security

In this example we treat noninterference security for a program fragment, based on the mathematical structure of Hidden Markov Models [8, 11, 16].

Suppose we have a "secret" program variable $h$ of type $H$ whose value could be partly revealed by an assignment statement $v{:=}\, exp$ to a visible variable $v$ of type $V$, if expression $exp$ contains $h$. Although an attacker cannot see $h$, he can see $v$'s final value, and he knows the program code (i.e. he knows the text of $exp$).

Given some known initial distribution $\delta$ in $\mathbb{D}H$ of $h$, how do we express what the attacker learns by executing the assignment, and how might we quantify the resulting security vulnerability? As an example we define $\delta{=}\{\!|0,1,2|\!\}$ to be a distribution on $h$ in $H{=}\{0,1,2\}$, with $v{:=}\, h \,\mathbf{mod}\, 2$ assigning its parity to $v$ of type $V{=}\{0,1\}$.

The output distribution over $V$ that the attacker observes in variable $v$ is

$$\{\!|h{:}\,\delta \bullet exp|\!\}\ , \tag{4}$$

thus in our example $\{\!|h{:}\,\{\!|0,1,2|\!\} \bullet h \,\mathbf{mod}\, 2|\!\}$. It equals $0\ {}_{2/3}\!\oplus\ 1$, showing that the attacker will observe $v{=}0$ twice as often as $v{=}1$.

The attacker is however not interested in $v$ itself: he is interested in $h$. When he observes $v{=}1$ what he learns, and remembers, is that definitely $h{=}1$. But when $v{=}0$ he learns "less" because the (*a posteriori*) distribution of $h$ in that case is $\{\!|0,2|\!\}$. In that case he is still not completely sure of $h$'s value.

In our style, for the first case $v{=}1$ the *a posteriori* distribution of $h$ is given by the conditional distribution $\{\!|h{:}\,\{\!|0,1,2|\!\} \mid h \,\mathbf{mod}\, 2 = 1|\!\} = \{\!|1|\!\}$; in the second case it is however $\{\!|h{:}\,\{\!|0,1,2|\!\} \mid h \,\mathbf{mod}\, 2 = 0|\!\} = \{\!|0,2|\!\}$; and in general it would be $\{\!|h{:}\,\{\!|0,1,2|\!\} \mid h \,\mathbf{mod}\, 2 = v|\!\}$ where $v$ is the observed value, either 0 or 1.

If in the example the attacker forgets $v$ but remembers what he learned about $h$, then $2/3$ of the time he remembers that $h$ has distribution $\{\!|0,2|\!\}$, i.e. is equally likely to be 0 or 2; and $1/3$ of the time he remembers that $h$ has distribution $\{\!|1|\!\}$, i.e. is certainly 1. Thus what he remembers about $h$ is

$$\{\!|0,2|\!\} \quad {}_{2/3}\!\oplus \quad \{\!|1|\!\}\ , \tag{5}$$

which is a distribution of distributions. [12] In general, what he remembers about $h$ is the distribution of distributions $\Delta$ given by

$$\Delta \quad := \quad \{\!|v{:}\,\{\!|h{:}\,\delta \bullet exp|\!\} \bullet \{\!|h{:}\,\delta \mid exp{=}v|\!\}|\!\}\ , \tag{6}$$

because $v$ itself has a distribution, as we noted at (4) above; and then the *a posteriori* distribution $\{\!|h{:}\,\delta \mid exp{=}v|\!\}$ of $h$ is determined by that $v$. The attacker's lack of interest in $v$'s actual value is reflected in $v$'s not being free in (6).

We now show what the attacker can do with (6), his analysis $\Delta$ of the program's meaning: if he guesses optimally for $h$'s value, with what probability

[12] In other work, we call this a *hyperdistribution* [15–17].

will he be right? For $v{=}0$ he will be right only half the time; but for $v{=}1$ he will be certain. So overall his attack will succeed with probability $\frac{1}{2}\ {}_{2/3}\!\oplus 1 = \frac{2}{3}{\times}\frac{1}{2} + \frac{1}{3}{\times}1 = 2/3$, obtained from (5) by replacing the two distributions with the attacker's "best guess probability" for each, the maximum of the probabilities in those distributions. We say that the "vulnerability" in this example is $2/3$.

For *vulnerability* in general take (6), apply the "best guess" strategy and then average over the cases: it becomes $(\mathcal{E}\eta{:}\,\Delta \bullet (\mathbf{max}\, h{:}\,H \bullet \eta.h))$, that is the maximum probability in each of the "inner" distributions $\eta$ of $\Delta$, averaged according to the "outer" probability $\Delta$ itself assigns to each. [13]

It is true that (6) appears complex if all you want is the information-theoretic vulnerability of a single assignment statement. But a more direct expression for that vulnerability is not compositional for programs generally; we need $\Delta$-like semantics from which the vulnerability can subsequently be calculated, because they contain enough additional information for composition of meanings. We show elsewhere that (6) is necessary and sufficient for compositionality [15].

### 9.3 Comparison with conventional notation

Given the assignment statement $v{:=}\,exp$ as above, define random variables $F$ for the function $exp$ in terms of $h$, and $I$ for $h$ itself (again as a function of $h$, i.e. the identity).

Then we determine the observed output distribution of $v$ from the input distribution $\delta$ of $h$ by the push-forward of $F_*(\delta)$, from Sec. 5.3, of $F$ over $\delta$.

Then define function $g^\delta$, depending on $h$'s initial distribution $\delta$, that gives for any value of $v$ the conditioning of $\delta$ by the event $F{=}v$. That is $g^\delta(v){:=}\,\Pr(I|F{=}v)$ where the Pr on the right refers to $\delta$.

Finally, the output hyperdistribution (6) of the attacker's resulting knowledge of $h$ is given by the push-forward $g^\delta_*(F_*(\delta))$ of $g^\delta$ over $F_*(\delta)$ which, because composition distributes through push-forward, we can rewrite as $(g^\delta{\circ}F)_*(\delta)$.

An analogous treatment of (6) is given at (8) below, where superscript $\delta$ in $g^\delta$ here reflects the fact that $\delta$ is free in the inner comprehension there.

### 9.4 Comparison with *qualitative* noninterference security

In a qualitative approach [19, 20] we would suppose a *set* $H{:=}\,\{0,1,2\}$ of hidden initial possibilities for $h$, not a distribution of them; and then we would execute the assignment $v{:=}\,h\ \mathbf{mod}\ 2$ as before. An observer's deductions are described by the set of sets $\{\{0,2\},\{1\}\}$, a demonic choice between knowing $h{\in}\{0,2\}$ and knowing $h{=}1$. The general $v{:=}\,exp$ gives $\{v{:}\,\{h{:}\,H \bullet exp\} \bullet \{h{:}\,H \mid exp{=}v\}\}$, which is a qualitative analogue of (6). [14]

[13] This is the *Bayes Vulnerability* of $\Delta$ [23].

[14] Written conventionally that becomes $\{\{h{\in}H \mid exp{=}v\} \mid v{\in}\{exp \mid h{\in}H\}\}$, where the left- and right occurrences of "|" now have different meanings. And then what does the middle one mean?

With the (extant) Eindhoven algebra of *set* comprehensions, and some calculation, that can be rewritten

$$\{h\colon H \bullet \{h'\colon H \mid exp{=}exp'\}\} \ , \tag{7}$$

where $exp'$ is $exp[h\backslash h']$. It is the partition of $H$ by the function $(\boldsymbol{\lambda} h\colon H \bullet exp)$. Analogously, with the algebra of *distribution* comprehensions (see (9) below) we can rewrite (6) to

$$\{\!\{h\colon \delta \bullet \{\!\{h'\colon H \mid exp{=}exp'\}\!\}\}\!\} \tag{8}$$

The occurrence of (7) and others similar, in our earlier qualitative security work [18, App. A], convinced us that there should be a *probabilistic* notational analogue (8) reflecting those analogies of meaning. This report has described how that was made to happen.

## 10 Monadic structures and other related work

The structure of the Eindhoven notation is monadic: for distributions it is the Giry monad $\Pi$ on a category $\mathsf{Mes}$ of measurable spaces, with measurable maps as its morphisms [7]; for sets, it is the powerset monad $\mathbb{P}$ on $\mathsf{Set}$. That accounts for many similarities, among which is the resemblance between (7) and (8).

The functor $\mathbb{D}$ takes a base set (actually measure space) to distributions (actually, measures) on it; and $\mathbb{D}$ applied to an arrow is the push-forward $()_*$. The unit transformation $\eta(x) := \{\!\{x\}\!\}$ forms the point distribution, and the multiply transformation $\mu.\Delta := (\mathcal{E}\delta\colon \Delta \bullet \delta) = \mathcal{E}\Delta$ forms a weighted average of the distributions $\delta$ found within a distribution of distributions $\Delta$.

Similarly, functor $\mathbb{P}$ takes a set to the set of its subsets; and $\mathbb{P}$ applied to an arrow is the relational image. The unit transformation takes $x$ to singleton $\{x\}$, and multiply makes distributed union $(\bigcup x\colon X \bullet x) = \bigcup X$ from set of sets $X$.

There are also correspondences with monads in functional programming; and a number of functional-programming packages have been put together on that basis [22, 4, 12, 6]. The goal of those is mainly to enable probabilistic functional programming, except for the last one where the emphasis is also on a notation for reasoning.

There is an obvious connection with multisets, where the value associated with elements is a nonnegative integer, rather than a fraction (a probability) as here, and there is no one-summing requirement. There might thus be a more general notational treatment applying to sets, multisets and distributions all at once, if a unifying principle for conditioning can be found.

A notable example of other related work, but with a different background, is Hehner's *Probabilistic Perspective* [9]. A distribution there is an expression whose free variables range over a separately declared sample space: for each assignment of values to the free variables, the expression gives the probability of that assignment as an observation: thus for $n\colon \mathbb{N}^+$ the expression $2^{-n}$ is an example of a geometric distribution on the positive integers.

With a single new operator $\updownarrow$, for normalisation, and existing programming-like notations, Hehner reconstructs many familiar artefacts of probability theory

(including conditional distributions and *a posteriori* analyses), and convincingly demystifies a number of probability puzzles, including some of those treated here.

A strategic difference between our two approaches is (we believe) that Hehner's aim is in part to put elementary probability theory on a simpler, more rational footing; we believe he succeeds. In the sense of our comments in Sec. 1, he is working "forwards." As we hope Sec. 9 demonstrated, we started instead with existing probabilistic constructions (essentially Hidden Markov Models as we explain elsewhere [16]), as a program semantics for noninterference, and then worked backwards towards the Eindhoven quantifier notation. One of the senses in which we met Hehner "in the middle" is that we both identify discrete distributions as first-class objects, for Hehner a real-valued expression over free variables of a type and for us a function from that type into the reals.

In conventional probability theory that explicit treatment of distributions, i.e. giving them names and manipulating them, does not occur until one reaches either proper measures or Markov chains. For us it is (in spirit) the former; we believe part of Hehner's approach can be explained in terms of the latter.

A technical difference is our explicit treatment of free- and bound variables, a principal feature of the Eindhoven notation and one reason we chose it.

## 11 Summary and prospects

We have argued that Eindhoven-style quantifier notation simplifies many of the constructions appearing in elementary probability. As evidence for this we invite comparison of the single expression (8) with the paragraphs of Sec. 9.3.

There is no space here to give a comprehensive list of calculational identities; but we mention two of them as examples of how the underlying structure mentioned above (Sec. 10) generates equalities similar to those already known in the Eindhoven notation applied to sets.

One identity is the trading rule

$$\begin{array}{ll} & (\mathcal{E}s{:}\,\delta \mid rng' \bullet \{\!|\, exp' \,|\!\} \mid rng \bullet exp) \\ = & (\mathcal{E}s{:}\,\delta \mid rng' \times rng[s\backslash exp'] \bullet exp[s\backslash exp']) \ , \end{array} \tag{9}$$

so-called because it "trades" components of an inner quantification into an outer one. Specialised to defaults for *true* for *rng* and $s$ for $exp'$, it gives an alternative to Def. 11. An identity similar to this took us from (6) to (8).

A second identity is the one used in Sec. 8.4, that $(\mathcal{E}s{:}\,\delta; s'{:}\,\delta' \mid rng \bullet exp)$ equals $(\mathcal{E}s{:}\,\delta \mid (\mathcal{E}s'{:}\,\delta' \bullet rng) \bullet exp)$ when $s'$ is not free in $exp$. As noted there, this corresponds to a similar trading rule between set comprehension and existential quantification: both are notationally possible only because variable bindings are explicitly given *even when those variables are not used*. This is just what the Eindhoven style mandates.

The notations here generalise to (non-discrete) probability measures, i.e. even to non-elementary probability theory, again because of the monadic structure. For example the integral of a measurable function given as expression *exp* in a variable $s$ on a sample space $S$, with respect to a measure $\mu$, could conventionally

be written $\int exp\,\mu(\mathrm{d}s)$.[15] We write it however as$(\mathcal{E}s{:}\,\mu \bullet exp)$, and have access to (9)-like identities such as

$$(\mathcal{E}s{:}\,\{\!|s'{:}\,\mu \bullet exp'|\!\} \bullet exp) \quad = \quad (\mathcal{E}s'{:}\,\mu \bullet exp[s\backslash exp])\ .$$

(See App. A for how this would be written conventionally for measures.)

We ended in Sec. 9 with an example of how the notation improves the treatment of probabilistic computer programs, particularly those presented in a denotational-semantic style and based on *Hidden Markov Models* for quantitative noninterference security [11, 16]. Although the example concludes this report, it was the starting point for the work.

**Acknowledgements** Jeremy Gibbons identified functional-programming activity in this area, and shared his own recent work with us. Frits Vaandrager generously hosted our six-month stay at Radboud University in Nijmegen during 2011. The use of this notation for security (Sec. 9.2) was in collaboration with Annabelle McIver and Larissa Meinicke [15, 16, and others]. Roland Backhouse, Eric Hehner, Bart Jacobs and David Jansen made extensive and helpful suggestions; in particular Jansen suggested looking at continuous distributions (i.e. those given as a density function). Annabelle McIver gave strategic advice on the presentation. Finally, we thank the referees for their careful reading and useful comments.

## References

1. Steve Cheng. A crash course on the Lebesgue integral and measure theory. `www.gold-saucer.org/math/lebesgue/lebesgue.pdf`, downloaded Dec. 2011.
2. E.W. Dijkstra. *A Discipline of Programming.* Prentice-Hall, 1976.
3. E.W. Dijkstra and C.S. Scholten. *Predicate Calculus and Program Semantics.* Springer-Verlag, 1990.
4. Martin Erwig and Steve Kollmansberger. Probabilistic functional programming in Haskell. *Journal of Functional Programming*, 16:21–34, 2006.
5. D.H. Fremlin. *Measure Theory.* Torres Fremlin, 2000.
6. Jeremy Gibbons and Ralf Hinze. Just do it: simple monadic equational reasoning. In Manuel M. T. Chakravarty, Zhenjiang Hu, and Olivier Danvy, editors, *ICFP*, pages 2–14. ACM, 2011.
7. M. Giry. A categorical approach to probability theory. In *Categorical Aspects of Topology and Analysis*, volume 915 of *Lecture Notes in Mathematics*, pages 68–85. Springer, 1981.
8. J.A. Goguen and J. Meseguer. Unwinding and inference control. In *Proc. IEEE Symp on Security and Privacy*, pages 75–86. IEEE Computer Society, 1984.
9. Eric C. R. Hehner. A probability perspective. *Form. Asp. Comput.*, 23:391–419, July 2011.

[15] Or not? We say "could" because "[there] are a number of different notations for the integral in the literature; for instance, one may find any of the following: $\int_Y s\,\mathrm{d}\mu$, $\int_Y s(x)\,\mathrm{d}\mu$, $\int_Y s(x)\mu$, $\int_Y s(x)\mu(\,\mathrm{d}x)$, or even $\int_Y s(x)\,\mathrm{d}x\ldots$" [1].

10. C. Jones and G. Plotkin. A probabilistic powerdomain of evaluations. In *Proceedings of the IEEE 4th Annual Symposium on Logic in Computer Science*, pages 186–95, Los Alamitos, Calif., 1989. Computer Society Press.
11. D. Jurafsky and J.H. Martin. *Speech and Language Processing.* Prentice Hall International, 2000.
12. Oleg Kiselyov and Chung-Chieh Shan. Embedded probabilistic programming. In Walid Taha, editor, *Domain-Specific Languages*, volume 5658 of *Lecture Notes in Computer Science*, pages 360–384. Springer Berlin / Heidelberg, 2009.
13. E Kowalski. Measure and integral. `www.math.ethz.ch/~kowalski/measure-integral.pdf`, downloaded Dec. 2011.
14. A.K. McIver and C.C. Morgan. *Abstraction, Refinement and Proof for Probabilistic Systems.* Tech Mono Comp Sci. Springer, New York, 2005.
15. Annabelle McIver, Larissa Meinicke, and Carroll Morgan. Compositional closure for Bayes Risk in probabilistic noninterference. In *Proceedings of the 37th international colloquium conference on Automata, languages and programming: Part II*, ICALP'10, pages 223–235, Berlin, Heidelberg, 2010. Springer-Verlag.
16. Annabelle McIver, Larissa Meinicke, and Carroll Morgan. Hidden-Markov program algebra with iteration. At arXiv:1102.0333v1; to appear in Mathematical Structures in Computer Science in 2012, 2011.
17. Annabelle McIver, Larissa Meinicke, and Carroll Morgan. A Kantorovich-monadic powerdomain for information hiding, with probability and nondeterminism. In *Proc. Logic in Computer Science (LiCS)*, 2012.
18. Carroll Morgan. Compositional noninterference from first principles. *Formal Aspects of Computing*, pages 1–24, 2010. `//dx.doi.org/10.1007/s00165-010-0167-y`.
19. C.C. Morgan. *The Shadow Knows:* Refinement of ignorance in sequential programs. In T. Uustalu, editor, *Math Prog Construction*, volume 4014 of *Springer*, pages 359–78. Springer, 2006. Treats *Dining Cryptographers.*
20. C.C. Morgan. *The Shadow Knows:* Refinement of ignorance in sequential programs. *Science of Computer Programming*, 74(8):629–653, 2009. Treats *Oblivious Transfer.*
21. C.C. Morgan, A.K. McIver, and K. Seidel. Probabilistic predicate transformers. *ACM Trans Prog Lang Sys*, 18(3):325–53, May 1996. `doi.acm.org/10.1145/229542.229547`.
22. Norman Ramsey and Avi Pfeffer. Stochastic lambda calculus and monads of probability distributions. *SIGPLAN Not.*, 37:154–165, January 2002.
23. G. Smith. Adversaries and information leaks (Tutorial). In G. Barthe and C. Fournet, editors, *Proc. 3rd Symp. Trustworthy Global Computing*, volume 4912 of *LNCS*, pages 383–400. Springer, 2007.

# Appendices

## A Measure spaces

More general than discrete distributions, *measures* are used for probability over infinite sample spaces, where expected value becomes integration [5]. Here we sketch how "measure comprehensions" might appear; continuous distributions would be a special case of those.

In Riemann integration we write $\int_a^b x^2 \,\mathrm{d}x$ for the integral of the real-valued squaring-function $sqr := (\boldsymbol{\lambda} x \bullet x^2)$ over the interval $[a,b]$, and in that notation the $x$ in $x^2$ is bound by the quantifier $\mathrm{d}x$. The scope of the binding is from $\int$ to $\mathrm{d}x$.

In Lebesgue integration however we write $\int sqr \,\mathrm{d}\mu$ for the integral of that same function over a measure $\mu$.

The startling difference between those two notations is the use of of the concrete syntax "d" that in Riemann integration's $\mathrm{d}x$ binds $x$, while for measures the $\mu$ in $\mathrm{d}\mu$ is free. To integrate the expression form of the squaring-function over $\mu$ we have to bind its $x$ in another way: two typical approaches are $\int x^2 \,\mu(\mathrm{d}x)$ and $\int x^2 \,\mathrm{d}\mu(x)$ [1]. [16]

An alternative is to achieve some uniformity by using $\mathrm{d}(\cdot)$ in the same way for both kinds of integrals [14]. We use $\int exp \,\mathrm{d}x$ for $\int (\boldsymbol{\lambda} x \bullet exp)$ in all cases; and the measure, or the bounds, are always found *outside* the expression, next to the integral sign $\int$. Thus we write $\int_\mu(\cdot)$ for integration over general measure $\mu$, and then the familiar $\int_a^b(\cdot)$ is simply a typographically more attractive presentation of the special case $\int_{[a,b]}(\cdot)$ over the uniform measure on the real interval $[a,b]$. [17]

Then with $f := (\boldsymbol{\lambda} x \bullet F)$ we would have the equalities

$$\int_{\{\!|c|\!\}} F \,\mathrm{d}x \quad = \quad \int_{\{\!|c|\!\}} f \quad = \quad f.c \qquad \text{one-point rule}$$

and

$$\int_{g_*.\mu} F \,\mathrm{d}x \quad = \quad \int_{g_*.\mu} f \quad = \quad \int_\mu f \circ g \qquad \begin{array}{l}\text{recall push-forward}\\ \text{from Sec. 5.3.}\end{array}$$

In the second case we equate the integral of $f$, over an (unnamed) measure formed by pushing function $g$ forward over measure $\mu$, with the integral of the functional composition $f \circ g$ over measure $\mu$ directly.

For complicated measures, unsuitable as subscripts, an alternative for the integral notation $\int_\mu exp \,\mathrm{d}x$ is the expected value $(\mathcal{E}x\colon \mu \bullet exp)$. The one-point rule is then written $(\mathcal{E}x\colon \{\!|exp|\!\} \bullet F) = F[x \backslash exp]$. In the second case we have

$$(\mathcal{E}x\colon \{\!|y\colon \mu \bullet G|\!\} \bullet F) \quad = \quad (\mathcal{E}y\colon \mu \bullet F[x \backslash G]) \;, \tag{10}$$

[16] And there are more, since "[if] we want to display the argument of the integrand function, alternate notations for the integral include $\int_{x \in X} f(x) \,\mathrm{d}\mu \ldots$" [13].

[17] This is more general than probabiity measures, since the (e.g. Lebesgue) measure $b - a$ of the whole interval $[a,b]$ can exceed one.

where function $g$ has become the lambda abstraction $(\boldsymbol{\lambda} y \bullet G)$. In Lem. 3 below we prove (10) for the discrete case.

## B Exploiting non-freeness in the constructor

Here we prove the nontrivial step referred forward from Sec. 8.4: the main assumption is that $s'$ is not free in *exp*. But should $\delta$ itself be an expression, we require that $s'$ not be free there either.

$$
\begin{array}{lll}
 & (\mathcal{E}s{:}\,\delta; s'{:}\,\delta' \mid rng \bullet exp) & \\
= & (\mathcal{E}s{:}\,\delta; s'{:}\,\delta' \bullet rng \times exp) \;/\; (\mathcal{E}s{:}\,\delta; s'{:}\,\delta' \bullet rng) & \text{“Def. 11”} \\
 & & \\
= & \dfrac{(\sum s{:}\,S; s'{:}\,S' \bullet \delta.s \times \delta'.s' \times rng \times exp)}{(\sum s{:}\,S; s'{:}\,S' \bullet \delta.s \times \delta'.s' \times rng)} & \text{“Def. 7”} \\
 & & \\
= & \dfrac{(\sum s{:}\,S \bullet \delta.s \times (\sum s'{:}\,S' \bullet \delta'.s' \times rng) \times exp)}{(\sum s{:}\,S \bullet \delta.s \times (\sum s'{:}\,S' \bullet \delta'.s' \times rng))} & \text{“}s'\text{ not free in }exp\text{”} \\
 & & \\
= & (\mathcal{E}s{:}\,\delta \bullet (\mathcal{E}s'{:}\,\delta' \bullet rng) \times exp) \;/\; (\mathcal{E}s{:}\,\delta \bullet (\mathcal{E}s'{:}\,\delta' \bullet rng)) & \text{“Def. 7”} \\
= & (\mathcal{E}s{:}\,\delta \mid (\mathcal{E}s'{:}\,\delta' \bullet rng) \bullet exp)\ . & \text{“Def. 11”}
\end{array}
$$

## C Assorted proofs related to definitions [18]

**Lemma 1.** *$\{\!\{s{:}\,\delta \bullet exp\}\!\}$ is a distribution on $\{s{:}\,\lceil\delta\rceil \bullet exp\}$*

*Proof:* We omit the simple proof that $0 \leq \{\!\{s{:}\,\delta \bullet exp\}\!\}$; for the one-summing property, we write $S$ for $\lceil\delta\rceil$ and calculate

$$
\begin{array}{lll}
 & (\sum e{:}\,\{s{:}\,S \bullet exp\} \bullet \{\!\{s{:}\,\delta \bullet exp\}\!\}.e) & \text{“let } e \text{ be fresh”} \\
= & (\sum e{:}\,\{s{:}\,S \bullet exp\} \bullet (\mathcal{E}s{:}\,\delta \bullet [exp{=}e])) & \text{“Def. 9”} \\
= & (\sum e{:}\,\{s{:}\,S \bullet exp\} \bullet (\sum s{:}\,S \bullet \delta.s \times [exp{=}e])) & \text{“Def. 7”} \\
= & (\sum s{:}\,S; e{:}\,\{s{:}\,S \bullet exp\} \bullet \delta.s \times [exp{=}e]) & \text{“merge and swap summations”} \\
= & (\sum s{:}\,S; e{:}\,\{s{:}\,S \bullet exp\} \mid exp{=}e \bullet \delta.s) & \text{“trading”} \\
= & (\sum s{:}\,S \bullet \delta.s) & \text{“one-point rule”} \\
= & 1\ . & \text{“}\delta\text{ is a distribution”}
\end{array}
$$

□

**Lemma 2.** *$\{\!\{s{:}\,\delta \mid rng\}\!\}$ is a distribution on $\lceil\delta\rceil$ if $(\mathcal{E}s{:}\,\delta \bullet rng) \neq 0$*

*Proof:* We omit the simple proof that $0 \leq \{\!\{s{:}\,\delta \mid rng\}\!\}$; for the one-summing property, we write $S$ for $\lceil\delta\rceil$ and calculate

[18] We thank Roland Backhouse for the suggestion to include the first two of these.

$$
\begin{array}{lll}
 & (\sum s'\colon S \bullet \{\!|s\colon\delta \mid rng|\!\}.s') & \text{“let } s' \text{ be fresh”} \\
= & (\sum s'\colon S \bullet (\mathcal{E}s\colon\delta \bullet rng \times [s{=}s'])\ /\ (\mathcal{E}s\colon\delta \bullet rng)) & \text{“Def. 10”} \\
= & (\sum s'\colon S \bullet (\mathcal{E}s\colon\delta \bullet rng \times [s{=}s']))\ /\ (\mathcal{E}s\colon\delta \bullet rng) & \text{“}s' \text{ not free in denominator”} \\
= & (\mathcal{E}s\colon\delta \bullet rng)\ /\ (\mathcal{E}s\colon\delta \bullet rng) & \text{“one-point rule; Def. 7”} \\
= & 1\ . & \text{“}(\mathcal{E}s\colon\delta \bullet rng){\neq}0\text{”}
\end{array}
$$

□

**Lemma 3.** $(\mathcal{E}x\colon\{\!|y\colon\delta \bullet G|\!\} \bullet F) = (\mathcal{E}y\colon\delta \bullet F[x\backslash G])$
This is Equation (10) in the discrete case.

*Proof:* Let $X$ be the support of $\{\!|y\colon\delta \bullet G|\!\}$, for which a more concise notation is given in App. D below, and let $Y$ be the support of $\delta$; we calculate

$$
\begin{array}{lll}
 & (\mathcal{E}x\colon\{\!|y\colon\delta \bullet G|\!\} \bullet F) & \\
= & (\sum x\colon X \bullet \{\!|y\colon\delta \bullet G|\!\}.x \times F) & \text{“Def. 7”} \\
= & (\sum x\colon X \bullet (\mathcal{E}y\colon\delta \bullet [G{=}x]) \times F) & \text{“Def. 9”} \\
= & (\sum x\colon X \bullet (\sum y\colon Y \bullet \delta.y \times [G{=}x]) \times F) & \text{“Def. 7”} \\
= & (\sum y\colon Y; x\colon X \bullet \delta.y \times [G{=}x] \times F) & \text{“distribution of summations”} \\
= & (\sum y\colon Y \bullet \delta.y \times F[x\backslash G]) & \text{“one-point rule for summation”} \\
= & (\mathcal{E}y\colon\delta \bullet F[x\backslash G])\ . & \text{“Def. 7”}
\end{array}
$$

□

From Lem. 3 we have immediately an analogous equality for distributions, since distribution comprehensions are a special case of expected values: a more succinct, point-free alternative to Def. 9 and Def. 13 is given by the equality

$$\{\!|s\colon\delta \mid rng \bullet exp|\!\} \quad = \quad (\mathcal{E}s\colon\delta \mid rng \bullet \{\!|exp|\!\})\ , \quad {}^{19} \tag{11}$$

where the right-hand expected value is being taken in a vector space (of discrete distributions). This is how we simplified (6) to (8) in Sec. 9.

## D Further identities

The identities below are motivated by the first one, i.e. Sec. D.1, justifying the idea that in a comprehension with both range and constructor one can think in terms of enforcing the range as a first step, and then the constructor to what results. The identities are listed in order of increasing generality.

For conciseness in this section we use $E^{new}_{old}$ for substitution and letters $R, E$ instead of words $rng, exp$ for expressions and $\lceil s\colon\delta \mid rng \bullet exp\rceil$ for the support $\lceil\{\!|s\colon\delta \mid rng \bullet exp|\!\}\rceil$.

[19] from $(\mathcal{E}s\colon\delta \mid rng \bullet \{\!|exp|\!\}).e = (\mathcal{E}s\colon\delta \mid rng \bullet \{\!|exp|\!\}.e) = (\mathcal{E}s\colon\delta \mid rng \bullet [exp{=}e])$ .

**D.1** ——————————

$$
\begin{array}{ll}
 & (\mathcal{E}s\colon \{\!\{ s\colon \delta \mid R \}\!\} \bullet E) \\
= & (\mathcal{E}e\colon \{\!\{ s\colon \delta \mid R \}\!\} \bullet E^e_s) \qquad \text{“fresh variable } e\text{”} \\
= & (\sum e\colon \lceil s\colon \delta \mid R \rceil \bullet \{\!\{ s\colon \delta \mid R \}\!\}.e \times E^e_s) \\
= & (\sum e\colon \lceil s\colon \delta \mid R \rceil \bullet (\mathcal{E}s\colon \delta \mid R \bullet [s{=}e]) \times E^e_s) \\
= & (\sum e\colon \lceil s\colon \delta \mid R \rceil \bullet (\mathcal{E}s\colon \delta \bullet R \times [s{=}e]) \times E^e_s/(\mathcal{E}s\colon \delta \bullet R)) \\
= & (\sum e\colon \lceil s\colon \delta \mid R \rceil \bullet \delta.e \times R^e_s \times E^e_s/(\mathcal{E}s\colon \delta \bullet R)) \qquad \text{“one-point rule”} \\
= & (\sum e\colon \lceil s\colon \delta \mid R \rceil \bullet \delta.e \times R^e_s \times E^e_s)/(\mathcal{E}s\colon \delta \bullet R) \qquad \text{“}e \text{ not free in } R \text{ or } \delta\text{”} \\
= & (\mathcal{E}e\colon \delta \bullet R^e_s \times E^e_s)/(\mathcal{E}s\colon \delta \bullet R) \qquad \text{“definition } \mathcal{E} \text{ and } \lceil s\colon \delta \mid R \rceil\text{”} \\
= & (\mathcal{E}s\colon \delta \bullet R \times E)/(\mathcal{E}s\colon \delta \bullet R) \qquad \text{“}e \text{ not free in } R, E\text{”} \\
= & (\mathcal{E}s\colon \delta \mid R \bullet E)\ . \qquad \text{“Def. 11”}
\end{array}
$$

**D.2** —————————— **D.1 for distributions**

$$
\begin{array}{ll}
 & \{\!\{ s\colon \{\!\{ s\colon \delta \mid R \}\!\} \bullet E \}\!\} \\
= & \{\!\{ s\colon \delta \mid R \bullet E \}\!\}\ . \qquad \text{“from Sec. D.1 under the same conditions, using (11)”}
\end{array}
$$

**D.3** ——————————

An elaboration of Sec. D.1 with constructor $F$, generalising Lem. 3.

$$
\begin{array}{ll}
 & (\mathcal{E}s\colon \{\!\{ s\colon \delta \mid R \bullet F \}\!\} \bullet E) \\
= & \qquad \text{“as for Sec. D.1…”} \\
 & (\sum e\colon \lceil s\colon \delta \mid R \bullet F \rceil \bullet (\mathcal{E}s\colon \delta \bullet R \times [F{=}e]) \times E^e_s/(\mathcal{E}s\colon \delta \bullet R)) \\
 & \\
= & \qquad \text{“… but cannot use one-point wrt. } F\text{”} \\
 & (\sum e\colon \lceil s\colon \delta \mid R \bullet F \rceil \bullet (\sum s\colon \lceil \delta \rceil \bullet \delta.s \times R \times [F{=}e]) \times E^e_s/(\mathcal{E}s\colon \delta \bullet R)) \\
= & (\sum s\colon \lceil \delta \rceil; e\colon \lceil s\colon \delta \mid R \bullet F \rceil \bullet \delta.s \times R \times [F{=}e] \times E^e_s/(\mathcal{E}s\colon \delta \bullet R)) \\
 & \\
= & (\sum s\colon \lceil \delta \rceil \bullet \delta.s \times R \times E^F_s/(\mathcal{E}s\colon \delta \bullet R)) \qquad \text{“if } \delta.s \text{ and } R \text{ both nonzero,} \\
 & \qquad\qquad \text{then } F{\in}\lceil s\colon \delta \mid R \bullet F \rceil; \\
 & \qquad\qquad e \text{ not free in } R\text{”} \\
= & (\sum s\colon \lceil \delta \rceil \bullet \delta.s \times R \times E^F_s)/(\mathcal{E}s\colon \delta \bullet R) \\
= & (\mathcal{E}s\colon \delta \bullet R \times E^F_s)/(\mathcal{E}s\colon \delta \bullet R) \\
= & (\mathcal{E}s\colon \delta \mid R \bullet E^F_s)\ .
\end{array}
$$

**D.4** —————————— **D.3 for distributions**

$$
\begin{array}{ll}
 & \{\!\{ s\colon \{\!\{ s\colon \delta \mid R \bullet F \}\!\} \bullet E \}\!\} \\
= & \{\!\{ s\colon \delta \mid R \bullet E^F_s \}\!\}\ . \qquad \text{“from Sec. D.3, under the same conditions”}
\end{array}
$$

**D.5** ——————————

An elaboration of Sec. D.3 with range $G$.

$$
\begin{array}{ll}
& (\mathcal{E}s\colon \{\!|\, s\colon \delta \mid R \bullet F \,|\!\} \mid G \bullet E) \\
= & (\mathcal{E}e\colon \{\!|\, s\colon \delta \mid R \bullet F \,|\!\} \mid G_s^e \bullet E_s^e) \\
= & (\mathcal{E}e\colon \{\!|\, s\colon \delta \mid R \bullet F \,|\!\} \bullet G_s^e \times E_s^e)/(\mathcal{E}e\colon \{\!|\, s\colon \delta \mid R \bullet F \,|\!\} \bullet G_s^e) \\
= & (\mathcal{E}s\colon \delta \mid R \bullet G_s^F \times E_s^F)/(\mathcal{E}s\colon \delta \mid R \bullet G_s^F) \qquad \text{“Sec. D.3”} \\
\\
= & \dfrac{(\mathcal{E}s\colon \delta \bullet R \times G_s^F \times E_s^F)/(\mathcal{E}s\colon \delta \bullet R)}{(\mathcal{E}s\colon \delta \bullet R \times G_s^F)/(\mathcal{E}s\colon \delta \bullet R)} \qquad \text{“if } (\mathcal{E}s\colon \delta \bullet R) \text{ nonzero”} \\
\\
= & (\mathcal{E}s\colon \delta \bullet R \times G_s^F \times E_s^F) \;/\; (\mathcal{E}s\colon \delta \bullet R \times G_s^F) \\
= & (\mathcal{E}s\colon \delta \mid R \times G_s^F \bullet E_s^F) \;.
\end{array}
$$

**D.6** —————————— **D.5 for distributions**

$$
\begin{array}{ll}
& \{\!|\, s\colon \{\!|\, s\colon \delta \mid R \bullet F \,|\!\} \mid G \bullet E \,|\!\} \\
= & \{\!|\, s\colon \delta \mid R \times G_s^F \bullet E_s^F \,|\!\} \;. \qquad \text{“from Sec. D.5, under the same conditions”}
\end{array}
$$

## E A special notation for kernel

Expression (8) suggests that distribution $\delta$ is partitioned into equivalence classes based on equality of elements $s\colon \lceil \delta \rceil$ wrt. the function $(\boldsymbol{\lambda} s \bullet \mathit{exp})$. For sets (i.e. without probability) this is a well-known construction that partitions a set, converting it into a set of pairwise-disjoint equivalence classes based on equality with respect to a function. Thus we propose

**Definition 14.** *Distribution kernel* The kernel of a distribution $\delta$ with respect to a range $\mathit{rng}$ in variable $s$ is

$$(\mathcal{K}s\colon \delta / \mathit{rng}) \quad := \quad \{\!|\, s\colon \Delta \bullet \{\!|\, s'\colon \Delta \mid \mathit{rng} = \mathit{rng}[s \backslash s'] \,|\!\} \,|\!\} \;.$$

□

Def. 14 gives a still more compact alternative $(\mathcal{K}h\colon \delta / \mathit{exp})$ for the effect of the assignment $v := \mathit{exp}$ on incoming distribution $\delta$ over hidden variable $h$.