A probabilistic theory of designs based on distributions

Riccardo Bresciani
Andrew Butterfield

Lero @ Trinity College Dublin

Motivation

It's all about probability!

- Reliability of programs
- Programs run on fallible hardware
- Probabilistic specifications
- Security
- ...

Chen and Sanders @ FM09

They talk of a "so-far-unachieved goal of unifying probabilism with other programming constructs in the style of Unifying Theories of Programming".

Main references

- Kozen's framework
- pGCL
- UTP — . . . but I believe that on the second day of UTP 2012 there is no need to give a general overview of UTP. ☺

We got a paper into *iFM2012*, with the title *A UTP semantics of pGCL as a homogeneous relation*.

UTP and the theory of designs

- Problem with the relational theory:

$$true ; \mathcal{P} \neq true$$

- Solution:

$$ok \wedge Pre \Rightarrow ok' \wedge Post$$

or, for short,

$$Pre \vdash Post$$

A distributional framework (I)

Main elements

- States: $\sigma : \underline{\mathcal{V}} \rightarrow \underline{\mathcal{W}}$
- Distributions: $\chi : \mathcal{S} \rightarrow \mathbb{R}$
- Weighting distributions: $\pi : \mathcal{S} \rightarrow [0..1]$
- Probability distributions: $\delta : \mathcal{S} \rightarrow [0..1] \ \wedge \ \|\delta\| \leq 1$

where $\|\chi\| \triangleq \sum_{\sigma \in \mathsf{dom}(\chi)} \chi(\sigma)$

A distributional framework (II)

Notable distributions:

- Point distribution: $\|\eta_\sigma\| = 1 \ \wedge \ \eta_\sigma(\sigma) = 1$
- Empty distribution: $\forall \sigma \bullet \epsilon(\sigma) = 0$
- Unit distribution: $\forall \sigma \bullet \iota(\sigma) = 1$

Operating on distributions

- Pointwise addition and multiplication
- Scaling
- Restriction
- Remap

A distributional framework (III)

Restriction

- Given a boolean condition c, the distribution $\chi\langle c\rangle$ is equal to the distribution χ purged of all states where c does not hold true.
- Given a distribution ξ, the distribution $\chi\langle\xi\rangle$ is equal to the pointwise multiplication $\chi \circ \xi$.

Sometimes it is more convenient to see a multiplication as a restriction and viceversa:

- $\chi\langle c\rangle = \chi \circ \iota\langle c\rangle$
- $\delta\langle\pi\rangle = \delta \circ \pi$

A distributional framework (IV)

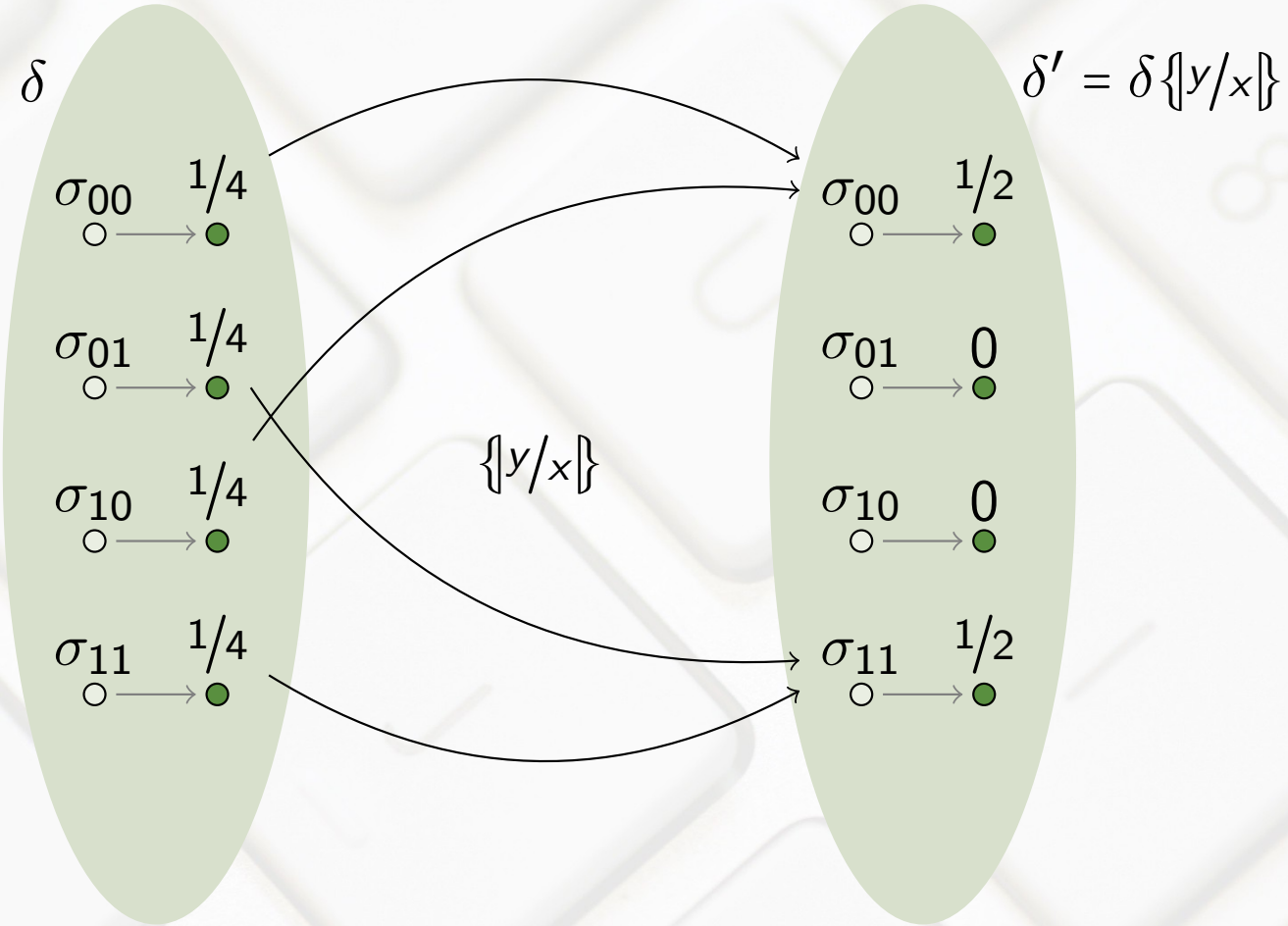
Remap

Given an expression $\underline{e}$ and a probability distribution δ, the remap operation $\delta\{\!|\underline{e}/\underline{v}|\!\}$ is defined as follows:

$$(\delta\{\!|\underline{e}/\underline{v}|\!\})(\sigma') \triangleq (\sum \delta(\sigma) \mid \sigma' = \sigma \dagger \{\underline{v} \mapsto \mathsf{eval}_\sigma(\underline{e}))$$

This is what happens when we perform the assignment $\underline{v} := \underline{e}$.

A distributional framework (V)



$$(\delta\{\!|y/x|\!\})(\sigma') \triangleq (\sum \delta(\sigma) \mid \sigma' = \sigma \dagger \{x \mapsto \mathsf{eval}_\sigma(y)\})$$

Probabilistic programs, UTP-style (I)

UTP semantics

$$
\begin{array}{rcl}
\mathit{abort} & \widehat{=} & \|\delta'\| \leq \|\delta\| \\
\mathit{miracle} & \widehat{=} & (\delta = \epsilon) \wedge (\delta' = \epsilon) \\
\mathit{skip} & \widehat{=} & \delta' = \delta \\
\underline{v} := \underline{e} & \widehat{=} & \delta' = \delta\{\underline{e}/\underline{v}\} \\
A;B & \widehat{=} & \exists \delta_m \bullet A(\delta, \delta_m) \wedge B(\delta_m, \delta') \\
\mathit{choice}(A, B, \mathcal{X}) & \widehat{=} & \exists \pi, \delta_A, \delta_B \bullet \pi \in \mathcal{X} \wedge A(\delta\langle\pi\rangle, \delta_A) \wedge \\
 & & \quad B(\delta\langle\bar{\pi}\rangle, \delta_B) \wedge \delta' = \delta_A + \delta_B \\
c * A & \widehat{=} & \mu X \bullet \mathit{choice}\big((A;X), \mathit{skip}, \{\iota\langle c\rangle\}\big)
\end{array}
$$

Probabilistic programs, UTP-style (II)

The benefits of homogeneity

- we hide (some) complexity under the hood — we deal with objects which can be manipulated more easily at a higher level;
- easier to address theories based on homogeneous relations (before- and after-observations of the same type);
- sequential composition is straightforward.

We pay this with an increased complexity in some definitions and proofs, but we are rewarded with a more easily usable framework.

Choice operators: syntactic sugar

$$choice(A, B, \mathcal{X}) \quad \hat{=} \quad \exists \pi, \delta_A, \delta_B \bullet \pi \in \mathcal{X} \wedge A(\delta\langle\pi\rangle, \delta_A) \wedge B(\delta\langle\bar{\pi}\rangle, \delta_B) \wedge \delta' = \delta_A + \delta_B$$

- for $\mathcal{X} = \{\iota\langle c\rangle\}$ we have conditional choice:

$$A \lhd c \rhd B \quad = \quad choice(A, B, \{\iota\langle c\rangle\})$$

- for $\mathcal{X} = \{p \cdot \iota\}$ we have probabilistic choice:

$$A \ {}_p\!\oplus B \quad = \quad choice(A, B, \{p \cdot \iota\})$$

- for $\mathcal{X} = \mathcal{D}_w$ we have non-deterministic choice:

$$A \sqcap B \quad = \quad choice(A, B, \mathcal{D}_w)$$

- for $\mathcal{X} = \{\epsilon, \iota\}$ we have ordinary disjunction:

$$A \vee B \quad = \quad choice(A, B, \{\epsilon, \iota\})$$

Healthiness conditions

Dist1 *feasibility*: $\|\delta'\| \leq \|\delta\|$

Dist2 *monotonicity*:
$\mathcal{P}(\delta_1, \delta_1') \wedge \mathcal{P}(\delta_2, \delta_2') \wedge \delta_2 > \delta_1 \Rightarrow \delta_2' \geq \delta_1'$

Dist3 *scaling*:
$\forall a \in \mathbb{R}^+ \wedge \|a \cdot \delta\| \leq 1 \bullet \mathcal{P}(\delta, \delta') \Leftrightarrow \mathcal{P}(a \cdot \delta, a \cdot \delta')$

Dist4 *convexity*, consequence of the purely random non-deterministic model:

$$(\mathcal{P}_1 \sqcap \mathcal{P}_2)(\delta, \delta') \Rightarrow \delta' \geq \min\big(\mathcal{P}_1(\delta) \cup \mathcal{P}_2(\delta)\big)$$

This poses restrictions on the space of possible program images, which is strictly a subset of $\wp\mathcal{D}$.

UTP probabilistic designs (I)

Given a standard design $\mathcal{P}re \vdash \mathcal{P}ost$ we can easily derive the corresponding probabilistic design with the observation that $ok \equiv (\|\delta\| = 1)$:

$$\begin{aligned}\mathcal{P}re \vdash \mathcal{P}ost &\equiv ok \wedge \mathcal{P}re \Rightarrow ok' \wedge \mathcal{P}ost \\ &\equiv \|\delta\| = 1 \wedge \mathcal{P}re \Rightarrow \|\delta'\| = 1 \wedge \mathcal{P}ost \\ &\equiv \|\delta\langle\mathcal{P}re\rangle\| = 1 \Rightarrow \|\delta'\langle\mathcal{P}ost\rangle\| = 1\end{aligned}$$

UTP probabilistic designs (II)

Assignment

$$\begin{aligned} \underline{v} := \underline{e} &\mathrel{\hat{=}} true \vdash \delta' = \delta\{\underline{e}/\underline{v}\} \\ &\equiv ok \wedge true \Rightarrow ok' \wedge \delta\{\underline{e}/\underline{v}\} \\ &\equiv \|\delta\| = 1 \Rightarrow \|\delta'\| = 1 \wedge \delta' = \delta\{\underline{e}/\underline{v}\} \end{aligned}$$

This states that an assignment is a valid design only if the expression e is defined everywhere in the state space: in fact undefinedness of e causes $\delta\{\underline{e}/\underline{v}\}$ to be a sub-distribution and therefore $\underline{v} := e$ reduces to $false$.

UTP probabilistic designs (III)

Skip

$$
\begin{aligned}
skip &\triangleq true \vdash \delta' = \delta \\
&\equiv ok \wedge true \Rightarrow ok' \wedge \delta \\
&\equiv \|\delta\| = 1 \Rightarrow \|\delta'\| = 1 \wedge \delta' = \delta \\
&\equiv \|\delta\| = 1 \Rightarrow \delta' = \delta
\end{aligned}
$$

UTP probabilistic designs (IV)

Abort

$$
\begin{aligned}
abort &\mathrel{\hat{=}} false \vdash false \\
&\equiv ok \wedge false \Rightarrow ok' \wedge false \\
&\equiv false \Rightarrow false \\
&\equiv true \\
&\equiv false \Rightarrow ok' \wedge true \\
&\equiv ok \wedge false \Rightarrow ok' \wedge true \\
&\equiv false \vdash true
\end{aligned}
$$

UTP probabilistic designs (V)

Chaos

$$
\begin{aligned}
chaos &\equiv true \vdash abort_R \\
&\equiv ok \wedge true \Rightarrow ok' \wedge \|\delta\| \leq \|\delta'\| \\
&\equiv \|\delta\| = 1 \Rightarrow \|\delta'\| = 1 \wedge \|\delta\| \leq \|\delta'\| \\
&\equiv \|\delta\| = 1 \Rightarrow \|\delta'\| = 1 \\
&\equiv ok \Rightarrow ok' \\
&\equiv true \vdash true \\
&\hat{=} chaos
\end{aligned}
$$

UTP probabilistic designs (VI)

Miracle (I)

$$
\begin{aligned}
miracle &\triangleq true \vdash false \\
&\equiv ok \wedge true \Rightarrow ok' \wedge false \\
&\equiv ok \Rightarrow false \\
&\equiv \neg ok \\
&\equiv \neg(\|\delta\| = 1) \\
&\equiv \|\delta\| < 1
\end{aligned}
$$

Miracle (II)

$$
\begin{aligned}
\mathit{miracle} &\equiv \mathit{true} \vdash \mathit{miracle}_R \\
&\equiv \mathit{true} \vdash (\delta = \epsilon) \wedge (\delta' = \epsilon) \\
&\equiv \mathit{ok} \wedge \mathit{true} \Rightarrow \mathit{ok}' \wedge (\delta = \epsilon) \wedge (\delta' = \epsilon) \\
&\equiv \mathit{ok} \Rightarrow \mathit{ok} \wedge \mathit{ok}' \wedge (\delta = \epsilon) \wedge (\delta' = \epsilon) \\
&\equiv \mathit{ok} \Rightarrow (\|\delta\| = 1) \wedge \mathit{ok}' \wedge (\delta = \epsilon) \wedge (\delta' = \epsilon) \\
&\equiv \mathit{ok} \Rightarrow \mathit{false} \\
&\equiv \neg \mathit{ok} \\
&\equiv \neg(\|\delta\| = 1) \\
&\equiv \|\delta\| < 1
\end{aligned}
$$

Total correctness

All healthiness conditions deriving from the distributional framework (Dist1–Dist4) obviously hold here as well; with a small modification we can recast the notion of total correctness by restricting Dist1 to a variant Dist1-TC (which implies Dist1), stating that:

$$\|\delta\| = \|\delta'\|$$

This requires a program to terminate with the same probability p with which it has started:

$$\|\delta\| = p \wedge \mathcal{P}re \Rightarrow \|\delta'\| = p \wedge \mathcal{P}ost$$

Link with the standard theory (I)

Standard designs have observations ok, ok', σ, σ'.

$$\begin{aligned} ok, ok' &: \mathbb{B} \\ \sigma, \sigma' &: \mathcal{S} \end{aligned}$$

Probabilistic designs have observations δ, δ'

$$\delta, \delta' : \mathcal{S} \to [0, 1]$$

Informally we require the two approaches to yield the same results when we are dealing with point distributions, *i.e.* when the probability of being in a given state is 1.

Link with the standard theory (II)

We define the linking predicate L as:

$$L((\delta, \delta'), (\sigma, \sigma', ok, ok')) \quad \widehat{=} \quad ok \Leftrightarrow (\|\delta'\| = 1) \wedge ok' \Leftrightarrow (\|\delta'\| = 1) \\ \wedge \delta = \eta_\sigma \wedge \delta' = \eta_{\sigma'}$$

This linking predicate allows us to introduce the following *Galois connections*; first we define the weakest probabilistic design corresponding to a standard design $\mathcal{P}_S$:

$$\forall \sigma, \sigma', ok, ok' \bullet L((\delta, \delta'), (\sigma, \sigma', ok, ok')) \Rightarrow \mathcal{P}_S(\sigma, \sigma', ok, ok')$$

Analogously, the strongest standard design corresponding to a probabilistic design $\mathcal{P}_D$ is:

$$\exists \delta, \delta' \bullet L((\delta, \delta'), (\sigma, \sigma', ok, ok')) \wedge \mathcal{P}_D(\delta, \delta')$$

Link with the standard theory (III)

This linking predicate is a bit too strong, as it maps many interesting program constructs to abort.

So far we have seen standard designs as relations:

$$\mathcal{P}_S : \mathcal{S} \times \mathbb{B} \to \mathcal{S} \times \mathbb{B}$$

but in order to build a more useful link we turn to this other interpretation:

$$\mathcal{P}_{\wp S} : \mathcal{S} \times \mathbb{B} \to \wp\mathcal{S} \times \mathbb{B}$$

which maps a state to what we may term its program image $\mathcal{P}(\sigma)$:

$$\mathcal{P}(\sigma) = \{\sigma' \mid \mathcal{P}_S(\sigma, \sigma')\}$$

Link with the standard theory (IV)

This allows us to define the following linking predicate:

$$L_{\wp}((\delta,\delta'),(\sigma,\alpha',o\!k,o\!k')) \quad \hat{=} \quad o\!k \Leftrightarrow (\|\delta'\| = 1) \wedge o\!k' \Leftrightarrow (\|\delta'\| = 1) \\ \wedge \delta = \eta_\sigma \wedge \mathsf{supp}(\delta') = \alpha'$$

We can state the variants of the previous Galois connections as:

$$\forall \sigma,\alpha',o\!k,o\!k' \bullet L_{\wp}((\delta,\delta'),(\sigma,\alpha',o\!k,o\!k')) \Rightarrow \mathcal{P}_{\wp S}(\sigma,\alpha',o\!k,o\!k')$$

$$\exists \delta,\delta' \bullet L_{\wp}((\delta,\delta'),(\sigma,\alpha',o\!k,o\!k')) \wedge \mathcal{P}_{D}(\delta,\delta')$$

This link maps all constructs from the probabilistic theory to the correspondent ones in the standard theory of designs, with the exception of generic choice, which is mapped to the nondeterministic choice.

Link with the standard theory (V)

Strongest standard design derivation:

$$\exists \delta, \delta' \bullet L_{\wp}((\delta, \delta'), (\sigma, \alpha', ok, ok')) \wedge \mathcal{P}_D(\delta, \delta')$$

$$\equiv \exists \delta, \delta' \bullet ok \Leftrightarrow (\|\delta'\| = 1) \wedge ok' \Leftrightarrow (\|\delta'\| = 1) \wedge \delta = \eta_\sigma \wedge \mathsf{supp}(\delta') = \alpha' \wedge \mathcal{P}_D(\delta, \delta')$$

$$\equiv \exists \delta, \delta' \bullet ok \Leftrightarrow (\|\delta'\| = 1) \wedge ok' \Leftrightarrow (\|\delta'\| = 1) \wedge \delta = \eta_\sigma \wedge \mathsf{supp}(\delta') = \alpha' \wedge \big((\|\delta\| = 1) \Rightarrow \|\delta'\| = 1 \wedge \mathcal{P}_{DR}(\delta, \delta')\big)$$

$$\equiv \exists \delta, \delta' \bullet \delta = \eta_\sigma \wedge \mathsf{supp}(\delta') = \alpha' \wedge \big(ok \Rightarrow ok' \wedge \mathcal{P}_{DR}(\delta, \delta')\big)$$

$$\equiv \exists \delta' \bullet \mathsf{supp}(\delta') = \alpha' \wedge \big(ok \Rightarrow ok' \wedge \mathcal{P}_{DR}(\eta_\sigma, \delta')\big)$$

$$\equiv ok \Rightarrow ok' \wedge \mathcal{P}_{\wp SR}(\sigma, \alpha')$$

$$\equiv \mathcal{P}_{\wp S}(\sigma, \alpha')$$

Conclusion

We have talked about:

- probabilistic theory of designs, which relies on a UTP-style framework based on distributions over the state space;
- embedding of the standard UTP theory, by requiring guaranteed termination from all program constructs;
- different Galois connections for projections/retreats with different characteristics.

Thanks for your attention!

Any questions?