Nondeterminstic Discrete Stochastic Differential Dynamic Logic

Samuel Kim (sdkim1@andrew.cmu.edu)
15-424 Term Paper

December 6, 2019

$$\frac{\sigma(\rho(\alpha, P), p) \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} Q}{\langle\!\langle \alpha \rangle\!\rangle P \leq p \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} Q} \ (\mathsf{d}\mathcal{L}\text{-}\langle\!\langle\rangle\!\rangle)$$

Abstract

While there are many powerful dynamic logics for describing the behavior of stochastic hybrid systems, none of them have easy-to-use implementations readily available. We introduce Nondeterminstic Discrete Stochastic Differential Dynamic Logic (NDS$\mathsf{d}\mathcal{L}$), another dynamic logic for describing the behavior of stochastic hybrid systems, and a conservative extension of $\mathsf{d}\mathcal{L}$. What sets NDS$\mathsf{d}\mathcal{L}$ apart from its predecessors is that it can be fully compiled down to $\mathsf{d}\mathcal{L}$, where we can use tools like KeYmaera X to prove properties about it. We also provide a NDS$\mathsf{d}\mathcal{L}$-to-$\mathsf{d}\mathcal{L}$ compiler, and prove soundness properties about the translation.

1 Introduction

Many physical processes that have probabilistic behavior can be described by an appropriate stochastic hybrid system. Dynamic logics along with their proof calculi for describing the behavior of these hybrid systems exist, but the issue is that none of them have powerful tooling like we have for $\mathsf{d}\mathcal{L}$. To remedy this, we introduce Nondeterminstic Discrete Stochastic Differential Dynamic Logic (NDS$\mathsf{d}\mathcal{L}$).

NDS$\mathsf{d}\mathcal{L}$ is a dynamic logic that describes the behavior of stochastic hybrid systems where the probabilistic behavior is restricted to discrete probability distributions. While this may reduce the expressiveness of the dynamic logic, what we gain is something far greater in value: the ability to translate NDS$\mathsf{d}\mathcal{L}$ to $\mathsf{d}\mathcal{L}$. This is not an obvious fact! It may seem that probabilistic behavior cannot be expressed in the nondeterminstic behavior of $\mathsf{d}\mathcal{L}$, but we show that it can be done. This means that we can use existing tooling for $\mathsf{d}\mathcal{L}$ like KeYmaera X to reason about NDS$\mathsf{d}\mathcal{L}$ formulas. Furthermore, we have also implemented a NDS$\mathsf{d}\mathcal{L}$-to-$\mathsf{d}\mathcal{L}$ compiler, so users can immediately start proving NDS$\mathsf{d}\mathcal{L}$ theorems using KeYmaera X. We also prove that the translation is sound, so users can trust the compiler.

2 Related Work

Other dynamic logics that describe the behavior of stochastic hybrid programs include S$\mathsf{d}\mathcal{L}$, introduced in [Pla11], and NS$\mathsf{d}\mathcal{L}$, introduced in [HMP15]. NS$\mathsf{d}\mathcal{L}$ combines S$\mathsf{d}\mathcal{L}$ with the nondeterminstic operators in $\mathsf{d}\mathcal{L}$ to form a new dynamic logic that can describe both stochastic and nondeterminstic behavior. However, none of these logics are currently implemented in KeYmaera X, making them hard to use. The semantics of NDS$\mathsf{d}\mathcal{L}$ are inspired by NS$\mathsf{d}\mathcal{L}$ semantics, but we restrict ourselves to discrete probabilistic operators.

3 Definitions

Definition 1 (NDSd$\mathcal{L}$ Syntax). The syntax of NDSd$\mathcal{L}$ is:

$$\begin{aligned}
\text{Term } e_1, e_2 &::= x \mid c \mid -e_1 \mid e_1 + e_2 \mid e_1 - e_2 \mid e_1 \times e_2 \\
\text{Formula } P, Q &::= \text{true} \mid \text{false} \mid P \wedge Q \mid P \vee Q \mid P \to Q \mid P \leftrightarrow Q \mid \neg P \mid e_1 = e_2 \\
&\quad \mid e_1 < e_2 \mid e_1 \leq e_2 \mid e_1 > e_2 \mid e_1 \geq e_2 \mid e_1 \neq e_2 \mid \forall x P \mid \exists x P \\
&\quad \mid [\alpha]P \mid \langle\alpha\rangle P \mid \langle\!|\alpha|\!\rangle P \leq p \\
\text{Program } \alpha, \beta &::= x := e \mid x := * \mid x := \{p_1 : e_1, \ldots, p_n : e_n\} \mid ?P \mid \alpha; \beta \mid \alpha^* \mid \alpha^{*:p} \\
&\quad \mid \alpha \cup \beta \mid \bigoplus_{i=1}^{n} p_i \alpha_i \mid \{x' = f(x) \,\&\, P\}
\end{aligned}$$

where $p, c \in \mathbb{Q}$, $0 \leq p \leq 1$.

We have new syntax added to d$\mathcal{L}$:

- p represents a constant probability.
- $\langle\!|\alpha|\!\rangle P \leq p$ computes an upper bound on the probability that P holds after any run of α and compares it with p. Note that this is similar to $\langle\cdot\rangle$ in Sd$\mathcal{L}$, but we have this as a formula instead of as a term [Pla11].
- $x := \{p_1 : e_1, \ldots, p_n : e_n\}$ assigns x from a discrete probability distribution that takes on value e_i with probability p_i. This upper bound will not be tight in general.
- $\alpha^{*:p}$ is a probabilistic loop, and loops with probability p.
- $\bigoplus_{i=1}^{n} p_i \alpha_i$ is a probabilistic choice that runs α_i with probability p_i.

Note that unlike in Sd$\mathcal{L}$ and NSd$\mathcal{L}$, probabilistic choice is n-ary instead of binary. This makes programs easier to understand from a user's perspective: it's hard to see at first glance that $\frac{1}{4}\alpha \oplus (\frac{1}{3}\beta \oplus (\frac{1}{2}\gamma \oplus \frac{1}{2}\delta))$ runs α, β, γ, or δ with equal probability.

We could have also introduced $\langle\!|\alpha|\!\rangle P$ as a term (which would evaluate to an upper bound on the probability that P holds after any run of α), but then we would also need a way to turn a formula into a term, with value 1 if the formula is true and 0 otherwise. This can be done, but it makes translation to d$\mathcal{L}$ messy, so we choose to keep it in the formula level.

Definition 2 (NDSd$\mathcal{L}$ Semantics Preliminaries). Let State be the set of all states, where states map variables to reals as in d$\mathcal{L}$ [Pla08].
Let **stuck** $\notin$ State be a special "stuck state" that represents states that have failed a test in a program. (**stuck** is not a real state! **stuck** cannot map variables to reals!)
Let $\omega[\![e]\!]$ be the evaluation of term e in a state ω. This is defined as in d$\mathcal{L}$.
Let Formula be the set of all NDSd$\mathcal{L}$ formulas.
Let Program be the set of all NDSd$\mathcal{L}$ programs.
Let Var be the set of all variables.
Let **SP** be the set of all (discrete) probability distributions over State $\cup$ {**stuck**}.
Let Det(ω) be the deterministic state distribution of ω, so $\mathbf{P}[\nu = \omega] = 1$ for $\nu \sim \text{Det}(\omega)$.
Let f_μ be the probability mass function of $\mu \in \mathbf{SP}$.
Let $\alpha^0 = ?\text{true}$, $\alpha^{n+1} = \alpha^n; \alpha$ for $n \in \mathbb{N}$.

We provide a transition semantics for NDS$\mathsf{d}\mathcal{L}$ like $\mathsf{d}\mathcal{L}$ and NS$\mathsf{d}\mathcal{L}$, where we keep track of a (discrete) probability distribution of states.

Definition 3 (NDS$\mathsf{d}\mathcal{L}$ Program Semantics). Let $[\![\cdot]\!]^{\mathsf{d}\mathcal{L}}$ be $[\![\cdot]\!]$ for programs as defined in $\mathsf{d}\mathcal{L}$.
We define a transition relation $[\![\cdot]\!] : \text{Program} \to 2^{\mathbf{SP}\times\mathbf{SP}}$ as follows:

$$\begin{aligned}
[\![x := e]\!] &= \{(\mu,\nu) : f_\nu = \sum_{\omega\in\text{supp}(f_\mu)} f_\mu(\omega)\text{Det}(\pi_\omega) \\
&\quad \text{where } (\omega,\pi_\omega) \in [\![x := e]\!]^{\mathsf{d}\mathcal{L}} \\
&\quad \text{or } \pi_\omega = \textbf{stuck} \text{ if no such } \pi_\omega \text{ exists}\} \\
[\![x := *]\!] &= \{(\mu,\nu) : f_\nu = \sum_{\omega\in\text{supp}(f_\mu)} f_\mu(\omega)\text{Det}(\pi_\omega) \\
&\quad \text{where } (\omega,\pi_\omega) \in [\![x := *]\!]^{\mathsf{d}\mathcal{L}} \\
&\quad \text{or } \pi_\omega = \textbf{stuck} \text{ if no such } \pi_\omega \text{ exists}\} \\
[\![x := \{p_1 : e_1, \ldots, p_n : e_n\}]\!] &= [\![\bigoplus_{i=1}^{n} p_i\{x := e_i\}]\!] \qquad (\textstyle\sum p_i = 1) \\
[\![?P]\!] &= \{(\mu,\nu) : f_\nu = \sum_{\omega\in\text{supp}(f_\mu)} f_\mu(\omega)\text{Det}(\pi_\omega) \\
&\quad \text{where } \pi_\omega = \omega \text{ if } \omega \in [\![P]\!] \\
&\quad \text{or } \pi_\omega = \textbf{stuck} \text{ otherwise}\} \\
[\![\alpha;\beta]\!] &= \{(\mu,\nu) : (\mu,\xi) \in [\![\alpha]\!], (\xi,\nu) \in [\![\beta]\!] \text{ for some } \xi\} \\
[\![\alpha^*]\!] &= \{(\mu,\nu) : \exists n \in \mathbb{N}, (\mu,\nu) \in \alpha^n\} \\
&= \{(\mu,\nu) : (\xi_i,\xi_{i+1}) \in [\![\alpha]\!]\ \forall\, 0 \le i \le n-1 \\
&\quad \text{for some } n \in \mathbb{N}, \xi_i \text{ where } \xi_0 = \mu, \xi_n = \nu\} \\
[\![\alpha^{*:p}]\!] &= \{(\mu,\nu) : f_\nu = \sum_{i=0}^{\infty}(1-p)p^i f_{\xi_i} \text{ for some } (\xi_i,\xi_{i+1}) \in [\![\alpha]\!] \\
&\quad \text{where } \xi_0 = \mu \text{ if } p < 1 \text{ and } \nu = \text{Det}(\textbf{stuck}) \text{ otherwise }\} \\
[\![\alpha \cup \beta]\!] &= [\![\alpha]\!] \cup [\![\beta]\!] \\
[\![\bigoplus_{i=1}^{n} p_i\alpha_i]\!] &= \{(\mu,\nu) : f_\nu = \sum_{i=1}^{n} p_i f_{\xi_i} \text{ for some } (\mu,\xi_i) \in [\![\alpha_i]\!]\} \quad (\textstyle\sum p_i = 1) \\
[\![\{x' = f(x)\ \&\ P\}]\!] &= \{(\mu,\nu) : f_\nu = \sum_{\omega\in\text{supp}(f_\mu)} f_\mu(\omega)\text{Det}(\pi_\omega) \\
&\quad \text{where } \varphi(0) = \omega \text{ except at } x', \pi_\omega = \varphi(r) \text{ for some solution} \\
&\quad \varphi : [0,r] \to \text{State, for some } r > 0 \text{ where for all times } 0 \le z \le r, \\
&\quad \varphi(z) \in [\![x' = f(x) \wedge P]\!] \text{ or } \pi_\omega = \textbf{stuck} \text{ if no such } \pi_\omega \text{ exists}\}
\end{aligned}$$

Note that in the definition of $[\![\{x' = f(x)\ \&\ P\}]\!]$, we have $(\omega,\pi_\omega) \in [\![\{x' = f(x)\ \&\ P\}]\!]^{\mathsf{d}\mathcal{L}}$, except that P is a NDS$\mathsf{d}\mathcal{L}$ formula instead of a $\mathsf{d}\mathcal{L}$ formula.

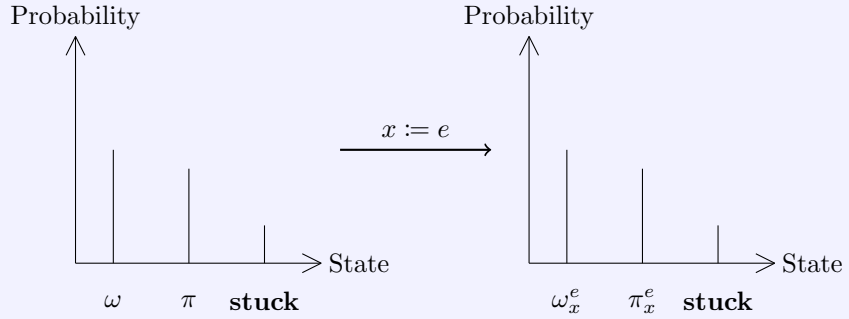
When we transition between state probability distributions for $\mathsf{d}\mathcal{L}$ hybrid programs, we update each state in the probability distribution according to the transition semantics for $\mathsf{d}\mathcal{L}$. Note that

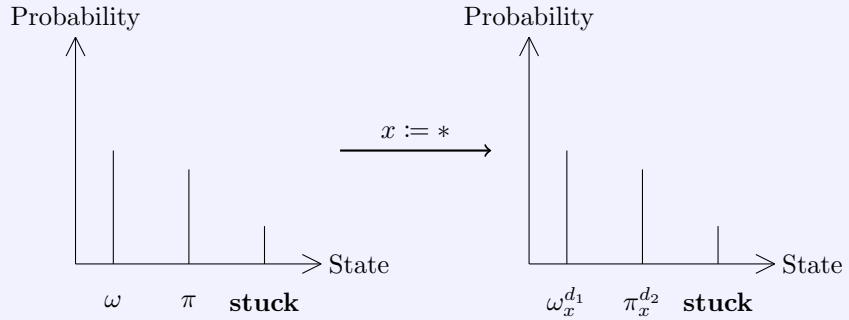
states can fail tests, in which case they are mapped to **stuck**. Note also that if two states map to the same state, their probabilities add.

Note that the probability distribution in the definition of $[\![\alpha^{*:p}]\!]$ is truly a probability distribution as it converges pointwise and the weights sum to 1. Note also that **stuck** in a probability distribution is always mapped to **stuck**.

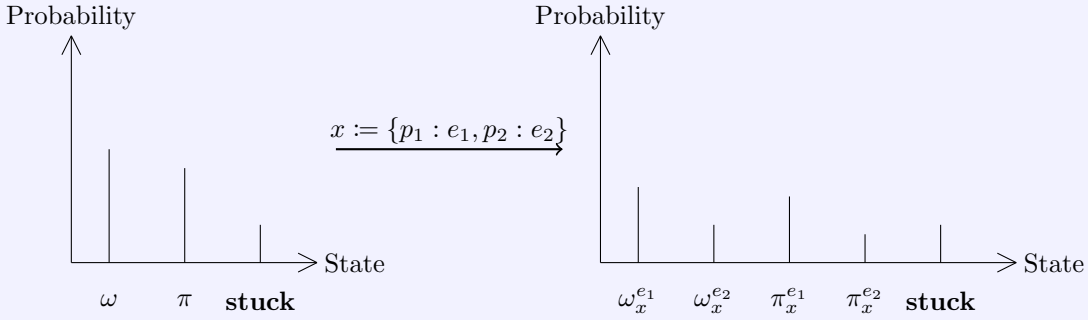
We illustrate these transitions below:

Assignments



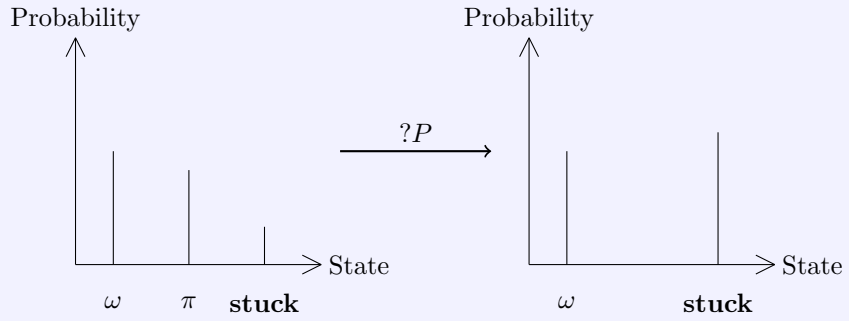


$d_1, d_2 \in \mathbb{R}$. Note that d_1 and d_2 can be different.



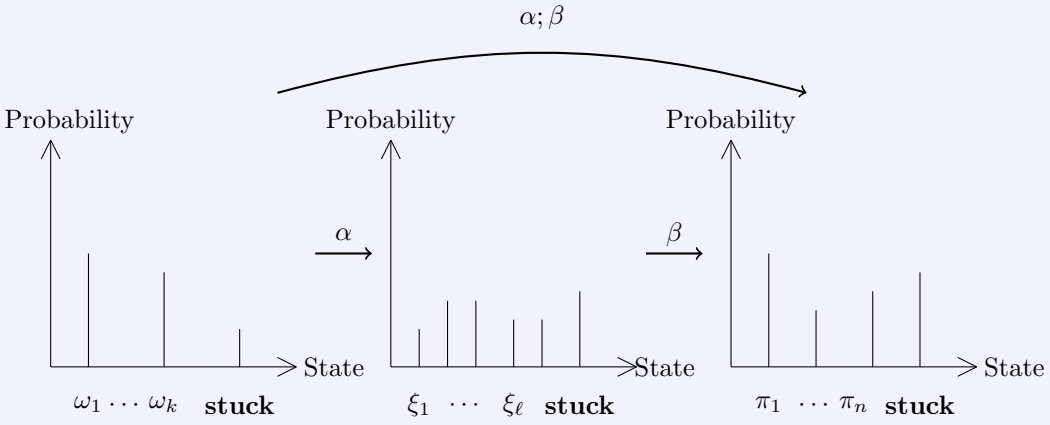
We can also view this as a probabilistic choice between $x := e_1$ and $x := e_2$.

Test

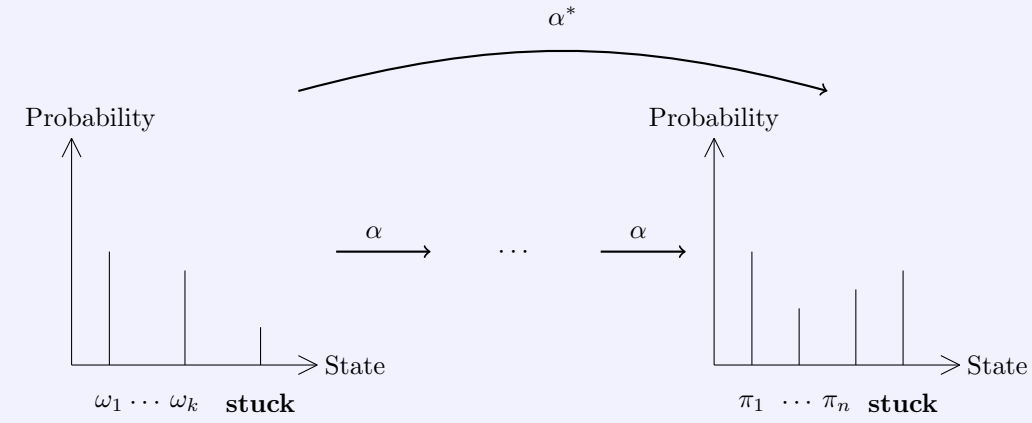


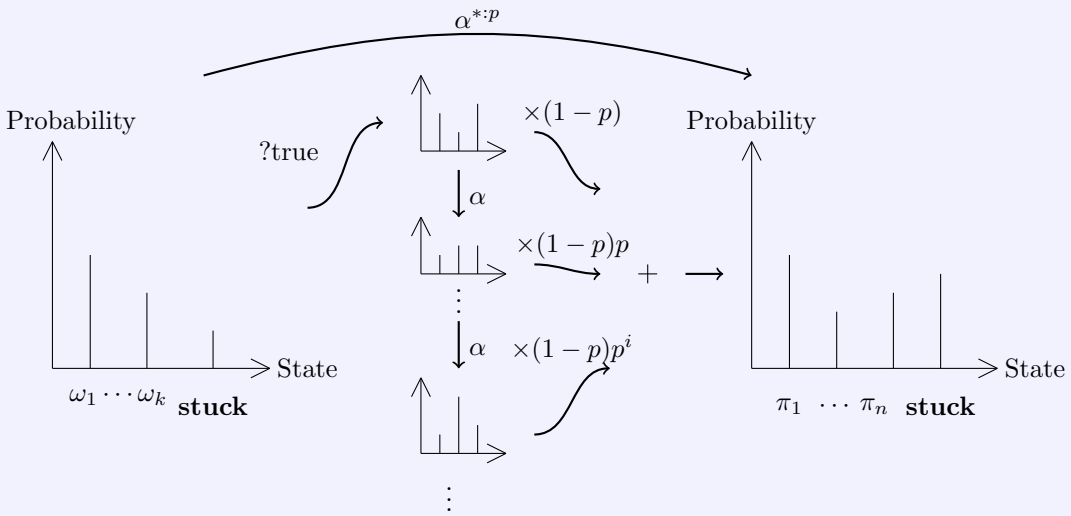
Here, state ω passed the test, while state π failed and was mapped to **stuck**.

Sequential Composition



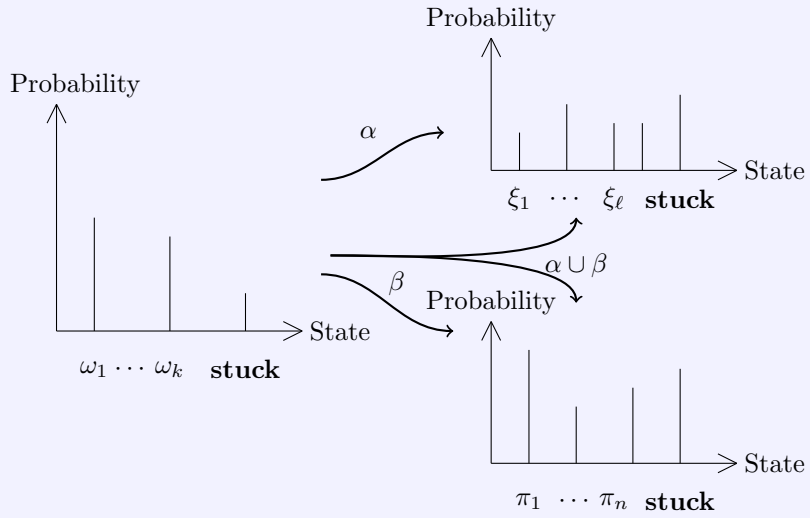
Repetition



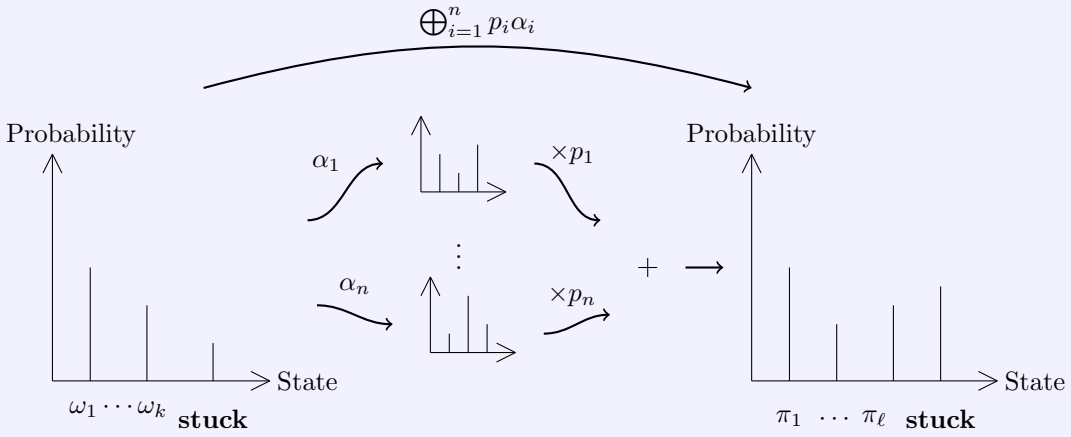


Note that we must transition from the previous iteration to get to the next iteration. We cannot directly transition from the starting distribution to every possible iteration.

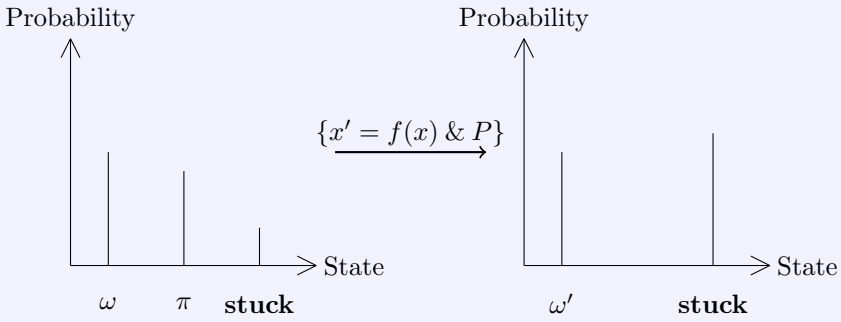
Choice



Both the α transition and the β transition are valid transitions for $\alpha \cup \beta$.



ODE



Here, state ω evolved to ω', while state π failed the test initially and was mapped to **stuck**.

Definition 4 ($\mathcal{I}$). $\mathcal{I}$ is the indicator function, and takes in a formula and a variable and sets the variable to the integer truth value of the formula.
$\mathcal{I} : \text{Formula} \times \text{Var} \to \text{Formula}$ is defined as

$$\mathcal{I}(P, y) = (P \wedge (y = 1)) \vee (\neg P \wedge (y = 0))$$

Definition 5 (ρ). ρ takes a program and a formula and outputs a set of preconditions and an upper bound on the probability that P is true in any run of the program.
We define $\rho : \text{Program} \times \text{Formula} \to 2^{\text{Formula}} \times \text{Term}$ as:

$$\begin{aligned}
\rho(\beta; x := e, P) &= \rho(\beta, \langle x := e\rangle P) \\
\rho(\beta; x := *, P) &= \rho(\beta, \langle x := *\rangle P) \\
\rho(\beta; x := \{p_1 : e_1, \ldots, p_n : e_n\}, P) &= \left(\bigcup_{i=1}^{n} X_i, \sum_{i=1}^{n} p_i f_i\right) \\
&\quad \text{where } (X_i, f_i) = \rho(\beta, \langle x := e_i\rangle P) \\
\rho(\beta; ?Q, P) &= \rho(\beta, \langle ?Q\rangle P) \\
\rho(\beta; \alpha^*, P) &= \rho(\beta, \langle \alpha^*\rangle P) \\
\rho(\beta; \alpha^{*:p}, P) &= \rho(\beta, \langle \alpha^{*:p}\rangle P) \\
\rho(\beta; \alpha_1 \cup \alpha_2, P) &= (X_1 \cup X_2, \max(e_1, e_2)) \\
&\quad \text{where } (X_i, e_i) = \rho(\beta; \alpha_i, P) \\
\rho\left(\beta; \bigoplus_{i=1}^{n} p_i \alpha_i, P\right) &= \left(\bigcup_{i=1}^{n} X_i, \sum_{i=1}^{n} p_i e_i\right) \\
&\quad \text{where } (X_i, e_i) = \rho(\beta; \alpha_i, P) \\
\rho\left(\beta; \{x' = f(x)\ \&\ Q\}, P\right) &= \rho(\beta, \langle \{x' = f(x)\ \&\ Q\}\rangle P) \\
\rho(\cdot, P) &= (\{\mathcal{I}(P, y)\}, y) \qquad (y \text{ fresh})
\end{aligned}$$

where β can be empty, resulting in the last case.
Note that ; is associative, so we can freely reassociate until we get one of these cases. Note also that the maximum function is not explicitly in $\mathsf{d}\mathcal{L}$ or NDS$\mathsf{d}\mathcal{L}$, but we can simulate it by adding more preconditions like in KeYmaera X.

Definition 6 (σ). σ packages up the output of ρ into a final formula, ready for further translation.
$\sigma : (2^{\text{Formula}} \times \text{Term}) \times \mathbb{Q} \to \text{Formula}$ is defined as

$$\sigma((\{P_1, \ldots, P_n\}, e), p) = \left(\bigwedge_{i=1}^{n} P_i\right) \to (e \leq p)$$

Definition 7 (NDSd$\mathcal{L}$ Formula Semantics). We define $[\![\cdot]\!] : \text{Formula} \to 2^{\text{State}}$, the set of states that satisfy a formula as follows:

$$
\begin{aligned}
[\![\text{true}]\!] &= \text{State} \\
[\![\text{false}]\!] &= \varnothing \\
[\![P \wedge Q]\!] &= [\![P]\!] \cap [\![Q]\!] \\
[\![P \vee Q]\!] &= [\![P]\!] \cup [\![Q]\!] \\
[\![P \to Q]\!] &= [\![P]\!]^{\complement} \cup [\![Q]\!] \\
[\![P \leftrightarrow Q]\!] &= [\![P \to Q]\!] \cap [\![Q \to P]\!] \\
[\![\neg P]\!] &= [\![P]\!]^{\complement} \\
[\![e_1 \sim e_2]\!] &= \{\omega : \omega[\![e_1]\!] \sim \omega[\![e_2]\!]\} && (\sim \in \{=, <, \leq, >, \geq, \neq\}) \\
[\![\forall x P]\!] &= \{\omega : \forall d \in \mathbb{R}, \omega_x^d \in [\![P]\!]\} \\
[\![\exists x P]\!] &= \{\omega : \exists d \in \mathbb{R}, \omega_x^d \in [\![P]\!]\} \\
[\![[\alpha]P]\!] &= \{\omega : \forall (\text{Det}(\omega), \nu) \in [\![\alpha]\!], \forall \pi \sim \nu \text{ where } \pi \neq \textbf{stuck}, \pi \in [\![P]\!]\} \\
[\![\langle\alpha\rangle P]\!] &= \{\omega : \exists (\text{Det}(\omega), \nu) \in [\![\alpha]\!], \exists \pi \sim \nu \text{ where } \pi \neq \textbf{stuck}, \pi \in [\![P]\!]\} \\
[\![\langle\!|\alpha|\!\rangle P \leq p]\!] &= [\![\sigma(\rho(\alpha, P), p)]\!]
\end{aligned}
$$

Note that for any formula P, $\textbf{stuck} \notin [\![P]\!]$.

4 Failed Alternate Approach

Instead of introducing $\langle\!|\alpha|\!\rangle P \leq p$, we could have introduced a special variable $\mathbf{p}$ that represents the probability of being in a particular path through the execution of a program. Some possible translation rules would be:

$$\frac{\alpha \overset{\text{d}\mathcal{L}}{\rightsquigarrow} \overline{\alpha} \qquad P \overset{\text{d}\mathcal{L}}{\rightsquigarrow} \overline{P}}{[\alpha]P \overset{\text{d}\mathcal{L}}{\rightsquigarrow} [\mathbf{p} := 1; \overline{\alpha}]\overline{P}} \ (\mathbf{p}\text{-d}\mathcal{L}\text{-}[]) \qquad \frac{\forall 1 \leq i \leq n,\ \alpha_i \overset{\text{d}\mathcal{L}}{\rightsquigarrow} \overline{\alpha_i}}{\bigoplus_{i=1}^{n} p_i \alpha_i \overset{\text{d}\mathcal{L}}{\rightsquigarrow} \{\mathbf{p} := p_1 \mathbf{p}; \overline{\alpha_1}\} \cup \cdots \cup \{\mathbf{p} := p_n \mathbf{p}; \overline{\alpha_n}\}} \ (\mathbf{p}\text{-d}\mathcal{L}\text{-}\oplus)$$

Note that at the beginning of every program run in a formula, we set $\mathbf{p}$ to 1, then $\mathbf{p}$ is modified by probabilistic choices, ignoring nondeterminism.

However, the problem with this approach is that the modalities are not very useful. We can only reason about $\mathbf{p}$ for every run or for some run, but we cannot reason about $\mathbf{p}$ as an aggregate over all runs.

For example, if we have a program where we toss 2 coins and count the number of heads:

$$\alpha \equiv h := 0; \{1/2\{h := h + 1\} \oplus 1/2\{?\text{true}\}; \{1/2\{h := h + 1\} \oplus 1/2\{?\text{true}\}\}$$

This program translates to

$$\overline{\alpha} \equiv h := 0; \{\mathbf{p} := p/2; h := h + 1 \cup \mathbf{p} := p/2; ?\text{true}\}; \{\mathbf{p} := p/2; h := h + 1 \cup \mathbf{p} := p/2; ?\text{true}\}$$

Note that $\mathbf{p}$ has the same value across all runs! If we wanted probability bounds on $h = 1$, the best we can do is

$$\langle \mathbf{p} := 1; \overline{\alpha} \rangle (h = 1 \wedge \mathbf{p} = 1/4)$$

to show that there is some run where $h = 1$, and that run has probability $1/4$.

We can then use the union bound to show that the probability that $h = 1$ is at least $1/4$, which is a very weak bound. This problem only gets worse with more probabilistic choices, where $\mathbf{p} \to 0$.

One thing we could do in this logic is to use $[\cdot]$ to show that a formula P (with translation $\overline{P}$) is true after running β (with translation $\overline{\beta}$) with probability 0 as follows:

$$[\mathbf{p} \coloneqq 1; \overline{\beta}](\mathbf{p} = 0 \vee \neg \overline{P})$$

However, the only way we can get probability 0 is if we have a probabilistic choice where one of the branches has probability 0, so it's already easy to detect if there are paths in the program that run with probability 0.

Thus, we instead use $\langle\!\langle \alpha \rangle\!\rangle P \leq p$ as our primary way of calculating probabilities that a formula is true after some run of a program.

5 NDS$\mathsf{d}\mathcal{L}$-to-$\mathsf{d}\mathcal{L}$ Translation

We define judgements $P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{P}$ and $\alpha \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\alpha}$ where P is a formula in NDS$\mathsf{d}\mathcal{L}$, α is a program in NDS$\mathsf{d}\mathcal{L}$, $\overline{P}$ is a formula in $\mathsf{d}\mathcal{L}$, and $\overline{\alpha}$ is a program in $\mathsf{d}\mathcal{L}$ by the following inference rules:

(Note that terms in NDS$\mathsf{d}\mathcal{L}$ are also terms in $\mathsf{d}\mathcal{L}$ so no translation is needed for terms.)

5.0.1 Formulas

$$\frac{}{\text{true} \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \text{true}}\ (\mathsf{d}\mathcal{L}\text{-True}) \qquad \frac{}{\text{false} \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \text{false}}\ (\mathsf{d}\mathcal{L}\text{-False}) \qquad \frac{P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{P}}{\neg P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \neg \overline{P}}\ (\mathsf{d}\mathcal{L}\text{-}\neg)$$

$$\frac{P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{P} \qquad Q \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{Q}}{P \wedge Q \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{P} \wedge \overline{Q}}\ (\mathsf{d}\mathcal{L}\text{-}\wedge) \qquad \frac{P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{P} \qquad Q \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{Q}}{P \vee Q \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{P} \vee \overline{Q}}\ (\mathsf{d}\mathcal{L}\text{-}\vee)$$

$$\frac{P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{P} \qquad Q \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{Q}}{P \to Q \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{P} \to \overline{Q}}\ (\mathsf{d}\mathcal{L}\text{-}\to) \qquad \frac{P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{P} \qquad Q \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{Q}}{P \leftrightarrow Q \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{P} \leftrightarrow \overline{Q}}\ (\mathsf{d}\mathcal{L}\text{-}\leftrightarrow)$$

$$\frac{}{e_1 = e_2 \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} e_1 = e_2}\ (\mathsf{d}\mathcal{L}\text{-}=) \qquad \frac{}{e_1 < e_2 \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} e_1 < e_2}\ (\mathsf{d}\mathcal{L}\text{-}<) \qquad \frac{}{e_1 \leq e_2 \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} e_1 \leq e_2}\ (\mathsf{d}\mathcal{L}\text{-}\leq)$$

$$\frac{}{e_1 > e_2 \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} e_1 > e_2}\ (\mathsf{d}\mathcal{L}\text{->}) \qquad \frac{}{e_1 \geq e_2 \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} e_1 \geq e_2}\ (\mathsf{d}\mathcal{L}\text{-}\geq) \qquad \frac{}{e_1 \neq e_2 \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} e_1 \neq e_2}\ (\mathsf{d}\mathcal{L}\text{-}\neq)$$

$$\frac{P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{P}}{\forall x P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \forall x \overline{P}}\ (\mathsf{d}\mathcal{L}\text{-}\forall) \qquad \frac{P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{P}}{\exists x P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \exists x \overline{P}}\ (\mathsf{d}\mathcal{L}\text{-}\exists) \qquad \frac{\alpha \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\alpha} \qquad P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{P}}{[\alpha]P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} [\overline{\alpha}]\overline{P}}\ (\mathsf{d}\mathcal{L}\text{-}[])$$

$$\frac{\alpha \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\alpha} \qquad P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{P}}{\langle\alpha\rangle P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \langle\overline{\alpha}\rangle\overline{P}}\ (\mathsf{d}\mathcal{L}\text{-}\langle\rangle) \qquad \frac{\sigma(\rho(\alpha, P), p) \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} Q}{\langle\!\langle \alpha \rangle\!\rangle P \leq p \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} Q}\ (\mathsf{d}\mathcal{L}\text{-}\langle\!\langle\rangle\!\rangle)$$

One concern is that if we have many probabilistic choices in a row while translating $\langle\!\langle \alpha \rangle\!\rangle P \leq p$, we get exponentially many formulas. However, this isn't a big problem, as we also have the same problem when trying to prove $\mathsf{d}\mathcal{L}$ formulas that use many choices in a row in a box or diamond modality in KeYmaera X, as we need to enumerate over all possible paths in the worst case.

5.0.2 Programs

Translation of programs converts all probabilistic operators into nondeterminstic operators. Note that this translation only happens within box and diamond modalities, and the modalities only care about events that happen with positive probability.

$$\frac{}{x := e \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} x := e}\ (\mathsf{d}\mathcal{L}\text{-}:=) \qquad \frac{}{x := * \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} x := *}\ (\mathsf{d}\mathcal{L}\text{-}:=*)$$

$$\frac{}{x := \{p_1 : e_1, \ldots, p_n : e_n\} \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \bigcup_{\substack{i=1 \\ p_i \neq 0}}^{n} x := e_i}\ (\mathsf{d}\mathcal{L}\text{-}:=\text{pmf}) \qquad \frac{P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{P}}{?P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} ?\overline{P}}\ (\mathsf{d}\mathcal{L}\text{-}?)$$

$$\frac{\alpha \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\alpha} \qquad \beta \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\beta}}{\alpha; \beta \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\alpha}; \overline{\beta}}\ (\mathsf{d}\mathcal{L}\text{-};) \qquad \frac{\alpha \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\alpha}}{\alpha^* \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\alpha}^*}\ (\mathsf{d}\mathcal{L}\text{-}*) \qquad \frac{}{\alpha^{*:0} \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} ?\text{true}}\ (\mathsf{d}\mathcal{L}\text{-}*\text{:}0)$$

$$\frac{\alpha \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\alpha}}{\alpha^{*:1} \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\alpha}^*; ?\text{false}}\ (\mathsf{d}\mathcal{L}\text{-}*\text{:}1) \qquad \frac{\alpha \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\alpha} \qquad 0 < p < 1}{\alpha^{*:p} \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\alpha}^*}\ (\mathsf{d}\mathcal{L}\text{-}*\text{:}p)$$

$$\frac{\alpha \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\alpha} \qquad \beta \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\beta}}{\alpha \cup \beta \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\alpha} \cup \overline{\beta}}\ (\mathsf{d}\mathcal{L}\text{-}\cup) \qquad \frac{\forall 1 \leq i \leq n,\ \alpha_i \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\alpha_i}}{\bigoplus_{i=1}^{n} p_i \alpha_i \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \bigcup_{\substack{i=1 \\ p_i \neq 0}}^{n} \overline{\alpha_i}}\ (\mathsf{d}\mathcal{L}\text{-}\oplus)$$

$$\frac{P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{P}}{\{x' = f(x)\ \&\ P\} \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \{x' = f(x)\ \&\ \overline{P}\}}\ (\mathsf{d}\mathcal{L}\text{-ODE})$$

6 Theorems

We have the following theorems (proofs of these theorems are in the appendices):

Theorem 1 (Soundness of Translation). Let $P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{P}$. P is valid in NDS$\mathsf{d}\mathcal{L}$ iff $\overline{P}$ is valid in $\mathsf{d}\mathcal{L}$.

This means that if we want to show that an NDS$\mathsf{d}\mathcal{L}$ formula is valid, it suffices to translate it into $\mathsf{d}\mathcal{L}$ and show that the resulting formula is valid. This is useful since once we have a $\mathsf{d}\mathcal{L}$ formula, we can use tools like KeYmaera X to aid our proof.

Theorem 2 (Probability bound). If $\langle\!|\alpha|\!\rangle P \leq p$ is true in state ω then $\forall(\text{Det}(\omega), \nu) \in [\![\alpha]\!]$, $\mathbf{P}[\pi \in [\![P]\!]] \leq p$ for $\pi \sim \nu$.

This theorem states that $\langle\!|\alpha|\!\rangle P \leq p$ really computes an upper bound for the probability of P being true after any run of α.

Theorem 3 (Conservative Extension). NDS$\mathtt{d}\mathcal{L}$ is a conservative extension of $\mathtt{d}\mathcal{L}$, i.e. every valid $\mathtt{d}\mathcal{L}$ formula is also a valid NDS$\mathtt{d}\mathcal{L}$ formula.

Note that when translating $\langle\!|\alpha|\!\rangle P \leq p$ for loops, we cannot use the following translation:

$$\rho(\beta; \alpha^*, P) = \left(\bigcup_{i=0}^{\infty} X_i, \sup_i e_i\right) \text{ where } (X_i, e_i) = \rho(\beta; \alpha^i, P) \text{ for } i \in \mathbb{N}$$

as we don't have a way to translate the supremum of an infinite set. Thus, we are forced to assume the worst case scenario where P is satisfied with probability 1 after some run of α, by using $\langle\cdot\rangle$ to see if such a run exists. Similarly, we have the same problem with probabilistic loops.

However, we can use the following theorems:

Theorem 4 (Loop unrolling).

$$[\![\alpha^*]\!] = [\![\{?\text{true}\} \cup \{\alpha; \alpha^*\}]\!]$$

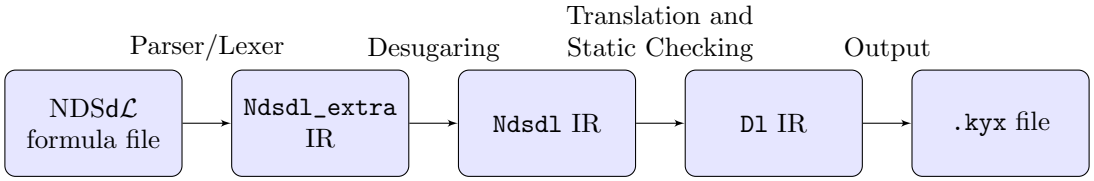
This lets us unroll nondeterminstic loops, so if we can show that the probability that P is true is bounded in the first few iterations, and P cannot be true in the other iterations, then we get a better bound on the probability that P is true.

Theorem 5 (Probabilistic loop unrolling).

$$[\![\alpha^{*:p}]\!] = [\![(1-p)\{?\text{true}\} \oplus p\{\alpha; \alpha^{*:p}\}]\!]$$

This lets us unroll probabilistic loops, so if we can show that P does not hold in the first few iterations of $\alpha^{*:p}$, then we can show that the probability that P holds is bounded by the probability of running the other iterations.

7 Implementation Details



We provide a compiler from NDS$\mathtt{d}\mathcal{L}$ to $\mathtt{d}\mathcal{L}$ in the supplementary material. The compiler is about 850 lines of OCaml, and takes in a NDS$\mathtt{d}\mathcal{L}$ formula from a file and outputs a `.kyx` file ready for use with KeYMaera X.

We have syntactic sugar for assigning a variable from a Bernoulli or (1-indexed) Geometric random variable as follows:

$$x := \text{Bernoulli}(p) \equiv x := \{p : 1, (1-p) : 0\}$$
$$x := \text{Geometric}(p) \equiv x := 1; \{x := x + 1\}^{*:(1-p)}$$

and syntactic sugar for allowing unrolling of probabilistic loops, by annotating the loop (or Geometric random variable) with `@unroll(n)`.

We also allow probabilities to be arbitrary terms as long as they are constant.

Just like in KeYMaera X, we also allow division and exponentiation of terms with the usual meaning.

7.1 Syntax

The compiler uses the following syntax:

```
Program a ::=
  | {p_1: a_1 +++ ... +++ p_n: a_n}     Probabilistic choice:
                                          p_i constant probabilities
                                          The choices are parsed as a list,
                                          not as binary choices.
  | {a}*:p                              Probabilistic loop:
                                          p constant probability
  | {a}*:p@unroll(n)                    Unroll probabilistic loop:
                                          p constant probability,
                                          n nonnegative integer.
                                          Unrolls the probabilistic loop n times
                                          for better probability bounds.
  | x := {p_1: e_1, ..., p_n: e_n}      Random assignment from pmf:
                                          p_i constant probabilities
  | x := Bernoulli(p)                   Bernoulli distribution:
                                          p constant probability
  | x := Geometric(p)                   (1-indexed) Geometric distribution:
                                          p constant probability
  | x := Geometric(p)@unroll(n)         Unroll (1-indexed) geometric distribution:
                                          p constant probability
                                          n nonnegative integer
  | a; b                                Note that the semicolon is explicitly
                                          required for sequential composition
                                          instead of after each assignment.
  | ?P                                  These have the same meaning as in
  | a ++ b                                KeYmaera X.
  | x := e
  | {a}*
  | {a}*@unroll(n)                      Unroll loop: n nonnegative integer
  | {x' = e, ...} | {x' = e, ... & P}
  | {a}

Term e ::=
  | x
```

```
  | c                                    Decimal literal.
  | -e | e_1+e_2 | e_1-e_2 | e_1*e_2 | e_1/e_2 | e_1^e_2 | (e)

Formula P ::=
  | true | false | P&Q | P|Q | P->Q | P<->Q | !P
  | e_1=e_2 | e_1<e_2 | e_1<=e_2 | e_1>e_2 | e_1>=e_2 | e_1!=e_2
  | \forall x e | \exists x e
  | [a]P | [a;]P                       We allow an extra semicolon at the end
  | <a>P | <a;>P                         of a program.
  | <|a|>P <= p | <|a;|>P <= p         Probability upper bound:
                                         p constant probability
  | (P)
```

7.2 Intermediate Representations

We use 3 intermediate representations:

- `Ndsdl_extra`: NDSd$\mathcal{L}$ with syntactic sugar for Bernoulli and Geometric random variables, and probabilistic loop unrolling.
- `Ndsdl`: Regular NDSd$\mathcal{L}$.
- `Dl`: Regular d$\mathcal{L}$.

7.3 Parser/Lexer

We use `ocamllex` and `menhir` to lex and parse user input as a `Ndsdl_extra` formula.

7.4 Translation and Static Checking

We translate from `Ndsdl_extra` to `Ndsdl` to get rid of the syntactic sugar, then translate from `Ndsdl` to `Dl` using the translation rules in section 5.

During translation, we also perform static checking:

- All probabilities are constants between 0 and 1.
- All probabilities in a probabilistic choice or discrete random variable assignment sum to 1.
- Unroll annotations have a nonnegative integer number of iterations.

7.5 Output

We output the `Dl` formula and all variables used into a user-specified `.kyx` file.

8 Examples

Flipping unfair coins Say we flip 2 coins with probability 1/4 of getting heads, and we want prove that the probability that we get 1 or 2 heads is bounded by 3/8.

We write the following NDSd$\mathcal{L}$ formula:

```
<|
  h := 0;
  {1/4: h := h+1 +++ 3/4: ?true};
```

```
  {1/4: h := h+1 +++ 3/4: ?true}
|>(h=1) <= 3/8
```

and run the compiler on it with output

```
ProgramVariables
  Real h;
  Real tt_0;
  Real tt_1;
  Real tt_2;
  Real tt_3;
End.

Problem
  (((((true) & ((((<h := 0;>(<h := (h)+(1);>(<h := (h)+(1);>((h) = (1))))) & ((tt_0) = (1))
      ) | ((!(<h := 0;>(<h := (h)+(1);>(<h := (h)+(1);>((h) = (1)))))) & ((tt_0) = (0)))))
       & ((((<h := 0;>(<?true;>(<h := (h)+(1);>((h) = (1))))) & ((tt_1) = (1))) | ((!(<h :=
       0;>(<?true;>(<h := (h)+(1);>((h) = (1)))))) & ((tt_1) = (0))))) & ((((<h := 0;>(<h
      := (h)+(1);>(<?true;>((h) = (1))))) & ((tt_2) = (1))) | ((!(<h := 0;>(<h := (h)+(1)
      ;>(<?true;>((h) = (1)))))) & ((tt_2) = (0))))) & ((((<h := 0;>(<?true;>(<?true;>((h)
      = (1))))) & ((tt_3) = (1))) | ((!(<h := 0;>(<?true;>(<?true;>((h) = (1)))))) & ((
      tt_3) = (0))))) -> ((((0)+(((1)/(4))*(((0)+(((1)/(4))*(tt_0)))+(((3)/(4))*(tt_1)))))
      +(((3)/(4))*(((0)+(((1)/(4))*(tt_2)))+(((3)/(4))*(tt_3))))) <= ((3)/(8)))
End.
```

We can then import it into KeYmaera X and prove it in one click with a single application of the `master` tactic.

Geometric distribution bounds Say we have a Geometric random variable x with $p = 3/4$, and we want to show that the probability that $x \geq 3$ is bounded by $1/16$.

We write the following NDSd$\mathcal{L}$ formula:

```
<| x := Geometric(3/4)@unroll(2) |>(x >= 3) <= 1/16
```

Note that we must use `@unroll(2)` to get better bounds on the probability, as otherwise we would calculate an upper bound of 1. We run the compiler on it with output

```
ProgramVariables
  Real tt_0;
  Real tt_1;
  Real tt_2;
  Real x;
End.

Problem
  ((((true) & (((<x := 1;>(<?true;>((x) >= (3)))) & ((tt_0) = (1))) | ((!(<x := 1;>(<?true
      ;>((x) >= (3))))) & ((tt_0) = (0))))) & ((((<x := 1;>(<x := (x)+(1);>(<?true;>((x) >=
       (3))))) & ((tt_1) = (1))) | ((!(<x := 1;>(<x := (x)+(1);>(<?true;>((x) >= (3)))))))
      & ((tt_1) = (0))))) & ((((<x := 1;>(<x := (x)+(1);>(<x := (x)+(1);>(<{x := (x)+(1)
      ;}*>((x) >= (3)))))) & ((tt_2) = (1))) | ((!(<x := 1;>(<x := (x)+(1);>(<x := (x)+(1)
      ;>(<{x := (x)+(1);}*>((x) >= (3))))))) & ((tt_2) = (0))))) -> ((((0)+(((1)-((1)-((3)
      /(4))))*(tt_0)))+(((1)-((3)/(4)))*(((0)+(((1)-((1)-((3)/(4))))*(tt_1)))+(((1)-((3)
      /(4)))*(tt_2))))) <= ((1)/(16)))
End.
```

and import it into KeYmaera X for proving. We need to do a little bit of work to show that the rest of the loop iterations don't contribute much to the final result, but the proof is fairly simple.

9 Discussion and Conclusion

We have presented NDS$\mathsf{d}\mathcal{L}$, a dynamic logic for describing the behavior of stochastic hybrid programs where we restrict all probabilistic operators to be discrete. While this may seem limiting, we also gain the ability to fully compile down to $\mathsf{d}\mathcal{L}$ so we can use tools like KeYmaera X to prove NDS$\mathsf{d}\mathcal{L}$ theorems. Furthermore, we have also presented a working compiler from NDS$\mathsf{d}\mathcal{L}$ to $\mathsf{d}\mathcal{L}$ and given examples of NDS$\mathsf{d}\mathcal{L}$ theorems that can be proven with the compiler, so NDS$\mathsf{d}\mathcal{L}$ is also practical.

In our project proposal, we initially followed the approach detailed in section 4. However, we found the resulting dynamic logic to not be very useful as described in the section, so we formulated $\langle\!|\alpha|\!\rangle P \leq p$ and its translation rules instead.

Future work involves extending NDS$\mathsf{d}\mathcal{L}$ to continuous probability distributions while still keeping the ability to compile down to $\mathsf{d}\mathcal{L}$.

10 Deliverables

A NDS$\mathsf{d}\mathcal{L}$ to $\mathsf{d}\mathcal{L}$ compiler is included in the supplementary material.

11 Acknowledgments

Thanks to Prof. Platzer for suggesting this project.

References

[HMP15] David Henriques, Paulo Mateus, and André Platzer. Nondeterministic stochastic differential dynamic logic. 2015.

[Pla08] André Platzer. Differential dynamic logic for hybrid systems. *J. Autom. Reasoning*, 41(2):143–189, 2008.

[Pla11] André Platzer. Stochastic differential dynamic logic for stochastic hybrid programs. In *Automated Deduction - CADE-23 - 23rd International Conference on Automated Deduction, Wroclaw, Poland, July 31 - August 5, 2011. Proceedings*, pages 446–460, 2011.

A Proof of Theorem 1

Let R be a NDS$\mathsf{d}\mathcal{L}$ formula, and let $R \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{R}$.

We will prove a stronger statement from which the theorem follows: $\omega \in [\![R]\!]$ iff $\omega \in [\![\overline{R}]\!]$ for any state ω.

(Note that states are the same between NDS$\mathsf{d}\mathcal{L}$ and $\mathsf{d}\mathcal{L}$, so we can make this statement. Note also that **stuck** is not a state.)

We induct on the last rule used in the derivation of $R \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{R}$.

A.1 d$\mathcal{L}$-True

We have $R = \text{true}$ and $\overline{R} = \text{true}$, so ω always satisfies R and $\overline{R}$. Thus, $\omega \in [\![R]\!]$ iff $\omega \in [\![\overline{R}]\!]$.

A.2 d$\mathcal{L}$-False

We have $R = \text{false}$ and $\overline{R} = \text{false}$, so ω never satisfies R and $\overline{R}$. Thus, $\omega \in [\![R]\!]$ iff $\omega \in [\![\overline{R}]\!]$.

A.3 d$\mathcal{L}$-$\neg$

We have $R = \neg P$ and $\overline{R} = \neg \overline{P}$, and by the inductive hypothesis we have $P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{P}$ where $\omega \in [\![P]\!]$ iff $\omega \in [\![\overline{P}]\!]$.

Now note that $[\![R]\!] = [\![P]\!]^{\complement}$ and $[\![\overline{R}]\!] = [\![\overline{P}]\!]^{\complement}$, so $\omega \in [\![R]\!]$ iff $\omega \in [\![\overline{R}]\!]$ as desired.

A.4 d$\mathcal{L}$-$\wedge$

We have $R = P \wedge Q$ and $\overline{R} = \overline{P} \wedge \overline{Q}$, and by the inductive hypothesis we have $P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{P}$, $Q \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{Q}$ where $[\![P]\!] = [\![\overline{P}]\!]$, $[\![Q]\!] = [\![\overline{Q}]\!]$.

Note that $[\![R]\!] = [\![P]\!] \cap [\![Q]\!] = [\![\overline{P}]\!] \cap [\![\overline{Q}]\!] = [\![\overline{R}]\!]$ as desired.

A.5 d$\mathcal{L}$-$\vee$

We have $R = P \vee Q$ and $\overline{R} = \overline{P} \vee \overline{Q}$, and by the inductive hypothesis we have $P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{P}$, $Q \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{Q}$ where $[\![P]\!] = [\![\overline{P}]\!]$, $[\![Q]\!] = [\![\overline{Q}]\!]$.

Note that $[\![R]\!] = [\![P]\!] \cup [\![Q]\!] = [\![\overline{P}]\!] \cup [\![\overline{Q}]\!] = [\![\overline{R}]\!]$ as desired.

A.6 d$\mathcal{L}$-$\rightarrow$

We have $R = P \rightarrow Q$ and $\overline{R} = \overline{P} \rightarrow \overline{Q}$, and by the inductive hypothesis we have $P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{P}$, $Q \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{Q}$ where $[\![P]\!] = [\![\overline{P}]\!]$, $[\![Q]\!] = [\![\overline{Q}]\!]$.

Note that $[\![R]\!] = [\![P]\!]^{\complement} \cup [\![Q]\!] = [\![\overline{P}]\!]^{\complement} \cup [\![\overline{Q}]\!] = [\![\overline{R}]\!]$ as desired.

A.7 d$\mathcal{L}$-$\leftrightarrow$

We have $R = P \leftrightarrow Q$ and $\overline{R} = \overline{P} \leftrightarrow \overline{Q}$, and by the inductive hypothesis we have $P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{P}$, $Q \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{Q}$ where $[\![P]\!] = [\![\overline{P}]\!]$, $[\![Q]\!] = [\![\overline{Q}]\!]$.

Note that $[\![R]\!] = [\![P \rightarrow Q]\!] \cap [\![Q \rightarrow P]\!] = [\![\overline{P} \rightarrow \overline{Q}]\!] \cap [\![\overline{Q} \rightarrow \overline{P}]\!] = [\![\overline{R}]\!]$ as desired.

A.8 d$\mathcal{L}$-$\sim$ ($\sim \in \{=, <, \leq, >, \geq, \neq\}$)

We have $R = e_1 \sim e_2$ and $\overline{R} = e_1 \sim e_2$.

Note that $[\![R]\!] = [\![e_1]\!] \sim [\![e_2]\!] = [\![\overline{R}]\!]$ as desired.

A.9 d$\mathcal{L}$-$\forall$

We have $R = \forall x P$ and $\overline{R} = \forall x \overline{P}$, and by the inductive hypothesis we have $P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{P}$ where $[\![P]\!] = [\![\overline{P}]\!]$.

Note that $[\![R]\!] = [\![\forall x P]\!] = \{\omega : \forall d \in \mathbb{R}, \omega_x^d \in [\![P]\!]\} = \{\omega : \forall d \in \mathbb{R}, \omega_x^d \in [\![\overline{P}]\!]\} = [\![\overline{R}]\!]$ as desired.

A.10 dℒ-∃

We have $R = \exists x P$ and $\overline{R} = \exists x \overline{P}$, and by the inductive hypothesis we have $P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{P}$ where $[\![P]\!] = [\![\overline{P}]\!]$.

Note that $[\![R]\!] = [\![\exists x P]\!] = \{\omega : \exists d \in \mathbb{R}, \omega_x^d \in [\![P]\!]\} = \{\omega : \exists d \in \mathbb{R}, \omega_x^d \in [\![\overline{P}]\!]\} = [\![\overline{R}]\!]$ as desired.

A.11 dℒ-[]

We have $R = [\alpha]P$ and $\overline{R} = [\overline{\alpha}]\overline{P}$, and by the inductive hypothesis we have $\alpha \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\alpha}$ and $P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{P}$ where $\omega \in [\![P]\!]$ iff $\omega \in [\![\overline{P}]\!]$.

Now note that

$$\begin{aligned}
[\![\neg\langle\alpha\rangle\neg P]\!] &= [\![\langle\alpha\rangle\neg P]\!]^{\complement} \\
&= \{\omega : \exists(\mathrm{Det}(\omega), \nu) \in [\![\alpha]\!], \exists \pi \sim \nu \text{ where } \pi \neq \textbf{stuck}, \pi \in [\![\neg P]\!]\}^{\complement} \\
&= \{\omega : \neg\exists(\mathrm{Det}(\omega), \nu) \in [\![\alpha]\!], \exists \pi \sim \nu \text{ where } \pi \neq \textbf{stuck}, \pi \in [\![\neg P]\!]\} \\
&= \{\omega : \neg\exists(\mathrm{Det}(\omega), \nu) \in [\![\alpha]\!], \exists \pi \sim \nu \text{ where } \pi \neq \textbf{stuck}, \pi \in [\![P]\!]^{\complement}\} \\
&= \{\omega : \neg\exists(\mathrm{Det}(\omega), \nu) \in [\![\alpha]\!], \exists \pi \sim \nu \text{ where } \pi \neq \textbf{stuck}, \neg\pi \in [\![P]\!]\} \\
&= \{\omega : \neg\neg\forall(\mathrm{Det}(\omega), \nu) \in [\![\alpha]\!], \forall \pi \sim \nu \text{ where } \pi \neq \textbf{stuck}, \pi \in [\![P]\!]\} \\
&= \{\omega : \forall(\mathrm{Det}(\omega), \nu) \in [\![\alpha]\!], \forall \pi \sim \nu \text{ where } \pi \neq \textbf{stuck}, \pi \in [\![P]\!]\} \\
&= [\![[\alpha]P]\!]
\end{aligned}$$

Then we can convert all boxes into diamonds, and use the other cases in the induction to show that $[\![\neg\langle\alpha\rangle\neg P]\!] = [\![\neg\langle\overline{\alpha}\rangle\neg\overline{P}]\!]$.

By an axiom in dℒ, we then have $[\![\neg\langle\overline{\alpha}\rangle\neg\overline{P}]\!] = [\![[\overline{\alpha}]\overline{P}]\!]$, so we have $[\![[\alpha]P]\!] = [\![[\overline{\alpha}]\overline{P}]\!]$ as desired.

A.12 dℒ-⟨⟩

We have $R = \langle\alpha\rangle P$ and $\overline{R} = \langle\overline{\alpha}\rangle\overline{P}$, and by the inductive hypothesis we have $\alpha \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\alpha}$ and $P \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{P}$ where $\omega \in [\![P]\!]$ iff $\omega \in [\![\overline{P}]\!]$.

We carry out an inner induction on the last rule used in the derivation of $\alpha \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\alpha}$.

A.12.1 dℒ-:=

We have $\alpha = x := e$ and $\overline{\alpha} = x := e$.

For the NDSdℒ formula, we have that $\omega \in [\![\langle x := e\rangle P]\!]$ iff $\exists(\mathrm{Det}(\omega), \nu) \in [\![x := e]\!]$, $\exists \pi \sim \nu$ where $\pi \neq \textbf{stuck}$, $\pi \in [\![P]\!]$.

Note that $\mathrm{supp}(f_{\mathrm{Det}(\omega)}) = \{\omega\}$, and $f_{\mathrm{Det}(\omega)}(\omega) = 1$.

Then if $(\mathrm{Det}(\omega), \nu) \in [\![x := e]\!]$, we must have $f_\nu = \mathrm{Det}(\pi_\omega)$ where $(\omega, \pi_\omega) \in [\![x := e]\!]^{\mathsf{d}\mathcal{L}}$ or $\pi_\omega = \textbf{stuck}$ if no such π_ω exists. But exactly 1 π_ω does exist: $\pi_\omega = \omega_x^e$.

Thus, $(\mathrm{Det}(\omega), \nu) \in [\![x := e]\!]$ iff $\nu = \mathrm{Det}(\omega_x^e)$, so $\omega \in [\![\langle x := e\rangle P]\!]$ iff $\omega_x^e \in [\![P]\!]$.

For the $\mathsf{d}\mathcal{L}$ formula, we have that $\omega \in [\![\langle x := e\rangle \overline{P}]\!]$ iff $\omega_x^e \in [\![\overline{P}]\!]$.

But note that $\omega_x^e \in [\![P]\!]$ iff $\omega_x^e \in [\![\overline{P}]\!]$.

Thus, $\omega \in [\![\langle x := e\rangle P]\!]$ iff $\omega \in [\![\langle x := e\rangle \overline{P}]\!]$ as desired.

A.12.2 $\mathsf{d}\mathcal{L}$-:=∗

We have $\alpha = x := *$ and $\overline{\alpha} = x := *$.

For the NDS$\mathsf{d}\mathcal{L}$ formula, we have that $\omega \in [\![\langle x := *\rangle P]\!]$ iff $\exists(\mathrm{Det}(\omega), \nu) \in [\![x := *]\!]$, $\exists \pi \sim \nu$ where $\pi \neq \mathbf{stuck}$, $\pi \in [\![P]\!]$.

Note that $\mathrm{supp}(f_{\mathrm{Det}(\omega)}) = \{\omega\}$, and $f_{\mathrm{Det}(\omega)}(\omega) = 1$.

Then if $(\mathrm{Det}(\omega), \nu) \in [\![x := *]\!]$, we must have $f_\nu = \mathrm{Det}(\pi_\omega)$ where $(\omega, \pi_\omega) \in [\![x := *]\!]^{\mathsf{d}\mathcal{L}}$ or $\pi_\omega = \mathbf{stuck}$ if no such π_ω exists. But such π_ω do exist: ω_x^d for some $d \in \mathbb{R}$. Thus, $\omega \in [\![x := *]\!]P$ iff $\omega_x^d \in [\![P]\!]$ for some $d \in \mathbb{R}$.

For the $\mathsf{d}\mathcal{L}$ formula, we have that $\omega \in [\![\langle x := *\rangle \overline{P}]\!]$ iff $\omega_x^d \in [\![\overline{P}]\!]$ for some $d \in \mathbb{R}$.

But note that $\omega_x^d \in [\![P]\!]$ iff $\omega_x^d \in [\![\overline{P}]\!]$.

Thus, $\omega \in [\![\langle x := *\rangle P]\!]$ iff $\omega \in [\![\langle x := *\rangle \overline{P}]\!]$ as desired.

A.12.3 $\mathsf{d}\mathcal{L}$-pmf

We have $\alpha = x := \{p_1 : e_1, \ldots, p_n : e_n\}$ and $\overline{\alpha} = \bigcup_{\substack{i=1 \\ p_i \neq 0}}^{n} x := e_i$.

Now note that $[\![x := \{p_1 : e_1, \ldots, p_n : e_n\}]\!] = [\![\bigoplus_{i=1}^{n} p_i\{x := e_i\}]\!]$ and $\bigoplus_{i=1}^{n} p_i\{x := e_i\} \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \bigcup_{\substack{i=1 \\ p_i \neq 0}}^{n} x := e_i$, so this case reduces to the cases $\mathsf{d}\mathcal{L}$-:= and $\mathsf{d}\mathcal{L}$-$\oplus$.

A.12.4 $\mathsf{d}\mathcal{L}$-?

We have $\alpha = ?Q$ and $\overline{\alpha} = ?\overline{Q}$ where $Q \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{Q}$ and $[\![Q]\!] = [\![\overline{Q}]\!]$ by the inductive hypothesis.

For the NDS$\mathsf{d}\mathcal{L}$ formula, we have that $\omega \in [\![\langle ?Q\rangle P]\!]$ iff $\exists(\mathrm{Det}(\omega), \nu) \in [\![?Q]\!]$, $\exists \pi \sim \nu$ where $\pi \neq \mathbf{stuck}$, $\pi \in [\![P]\!]$.

Note that $\mathrm{supp}(f_{\mathrm{Det}(\omega)}) = \{\omega\}$, and $f_{\mathrm{Det}(\omega)}(\omega) = 1$.

Then if $(\mathrm{Det}(\omega), \nu) \in [\![x := e]\!]$, we must have $f_\nu = \mathrm{Det}(\pi_\omega)$ where $\pi_\omega = \omega$ if $\omega \in [\![Q]\!]$ or $\pi_\omega = \mathbf{stuck}$ otherwise.

Thus, $\omega \in [\![\langle ?Q\rangle P]\!]$ iff $\omega \in [\![Q]\!]$ and $\omega \in [\![P]\!]$.

For the $\mathsf{d}\mathcal{L}$ formula, we have that $\omega \in [\![\langle ?\overline{Q}\rangle \overline{P}]\!]$ iff $\omega \in [\![\overline{Q}]\!]$ and $\omega \in [\![\overline{P}]\!]$.

But note that $[\![Q]\!] = [\![\overline{Q}]\!]$ and $[\![P]\!] = [\![\overline{P}]\!]$.

Thus, $\omega \in [\![\langle ?Q\rangle P]\!]$ iff $\omega \in [\![\langle ?\overline{Q}\rangle \overline{P}]\!]$ as desired.

A.12.5 $\mathsf{d}\mathcal{L}$-;

We have $\alpha = \beta;\gamma$ and $\overline{\alpha} = \overline{\beta};\overline{\gamma}$ where $\beta \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\beta}$, $\gamma \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\gamma}$, and $[\![\langle\beta\rangle P]\!] = [\![\langle\overline{\beta}\rangle\overline{P}]\!]$, $[\![\langle\gamma\rangle P]\!] = [\![\langle\overline{\gamma}\rangle\overline{P}]\!]$ by the inductive hypothesis.

For the NDS$\mathsf{d}\mathcal{L}$ formula, we have that $\omega \in [\![\langle\beta;\gamma\rangle P]\!]$ iff $\exists(\mathrm{Det}(\omega),\nu) \in [\![\beta;\gamma]\!]$, $\exists\pi \sim \nu$ where $\pi \neq \textbf{stuck}$, $\pi \in [\![P]\!]$.

This happens iff $\exists(\mathrm{Det}(\omega),\xi) \in [\![\beta]\!]$, $\exists(\xi,\nu) \in [\![\gamma]\!]$, $\exists\pi \sim \nu$ where $\pi \neq \textbf{stuck}$, $\pi \in [\![P]\!]$.

Now note that $\exists(\xi,\nu) \in [\![\gamma]\!]$, $\exists\pi \sim \nu$ where $\pi \neq \textbf{stuck}$, $\pi \in [\![P]\!]$ iff there exists some $\tau \in \mathrm{supp}(\xi)$ such that $\exists(\mathrm{Det}(\tau),\nu') \in [\![\gamma]\!]$ where $\pi \in \mathrm{supp}(\nu')$, since some state in ξ must map to π in ν.

Thus, $\omega \in [\![\langle\beta;\gamma\rangle P]\!]$ iff $\omega \in [\![\langle\beta\rangle\langle\gamma\rangle P]\!]$.

By the inductive hypothesis, we have $\omega \in [\![\langle\beta\rangle\langle\gamma\rangle P]\!]$ iff $\omega \in [\![\langle\overline{\beta}\rangle\langle\overline{\gamma}\rangle\overline{P}]\!]$.

For the $\mathsf{d}\mathcal{L}$ formula, we have that $\omega \in [\![\langle\overline{\beta};\overline{\gamma}\rangle\overline{P}]\!]$ iff $\omega \in [\![\langle\overline{\beta}\rangle\langle\overline{\gamma}\rangle\overline{P}]\!]$.

Thus, $\omega \in [\![\langle\beta;\gamma\rangle P]\!]$ iff $\omega \in [\![\langle\overline{\beta};\overline{\gamma}\rangle\overline{P}]\!]$ as desired.

A.12.6 $\mathsf{d}\mathcal{L}$-$*$

We have $\alpha = \beta^*$ and $\overline{\alpha} = \overline{\beta}^*$ where $\beta \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\beta}$, and $[\![\langle\beta\rangle P]\!] = [\![\langle\overline{\beta}\rangle\overline{P}]\!]$ by the inductive hypothesis.

For the NDS$\mathsf{d}\mathcal{L}$ formula, we have that $\omega \in [\![\langle\beta^*\rangle P]\!]$ iff $\exists(\mathrm{Det}(\omega),\nu) \in [\![\beta^*]\!]$, $\exists\pi \sim \nu$ where $\pi \neq \textbf{stuck}$, $\pi \in [\![P]\!]$.

This happens iff $n \in \mathbb{N}$, $\exists(\mathrm{Det}(\omega),\nu) \in [\![\beta^n]\!]$, $\exists\pi \sim \nu$ where $\pi \neq \textbf{stuck}$, $\pi \in [\![P]\!]$. But this means that $\omega \in [\![\langle\beta^*\rangle P]\!]$ iff $\exists n \in \mathbb{N}$ such that $\omega \in [\![\langle\beta^n\rangle P]\!]$.

We use the $\mathsf{d}\mathcal{L}$-; case and the fact that $[\![\langle\beta\rangle P]\!] = [\![\langle\overline{\beta}\rangle\overline{P}]\!]$ to show that $\exists n \in \mathbb{N}$, $\omega \in [\![\langle\beta^n\rangle P]\!]$ iff $\exists n \in \mathbb{N}$, $\omega \in [\![\langle\overline{\beta}^n\rangle\overline{P}]\!]$.

But $\exists n \in \mathbb{N}$, $\omega \in [\![\langle\overline{\beta}^n\rangle\overline{P}]\!]$ iff $\omega \in [\![\langle\overline{\beta}^*\rangle\overline{P}]\!]$.

Thus, $\omega \in [\![\langle\beta^*\rangle P]\!]$ iff $\omega \in [\![\langle\overline{\beta}^*\rangle\overline{P}]\!]$ as desired.

A.12.7 $\mathsf{d}\mathcal{L}$-$*$:0

We have $\alpha = \beta^{*:0}$ and $\overline{\alpha} = ?\mathrm{true}$ where $\beta \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\beta}$, and $[\![\langle\beta\rangle P]\!] = [\![\langle\overline{\beta}\rangle\overline{P}]\!]$ by the inductive hypothesis.

For the NDS$\mathsf{d}\mathcal{L}$ formula, we have that $\omega \in [\![\langle\beta^{*:0}\rangle P]\!]$ iff $\exists(\mathrm{Det}(\omega),\nu) \in [\![\beta^{*:0}]\!]$, $\exists\pi \sim \nu$ where $\pi \neq \textbf{stuck}$, $\pi \in [\![P]\!]$.

Now note that we must have $f_\nu = \sum_{i=0}^{\infty}(1-0)0^i f_{\xi_i} = f_{\xi_0}$ where $\xi_0 = \mathrm{Det}(\omega)$. Thus, $f_\nu = f_{\mathrm{Det}(\omega)}$, so $\nu = \mathrm{Det}(\omega)$.

Thus, $\omega \in [\![\langle\beta^{*:0}\rangle P]\!]$ iff $\omega \in [\![P]\!]$.

For the $\mathsf{d}\mathcal{L}$ formula, we have that $\omega \in [\![\langle?\mathrm{true}\rangle\overline{P}]\!]$ iff $\omega \in [\![\overline{P}]\!]$.

Thus, $\omega \in [\![\langle\beta^{*:0}\rangle P]\!]$ iff $\omega \in [\![\langle?\mathrm{true}\rangle\overline{P}]\!]$ as desired.

A.12.8 $\mathsf{d}\mathcal{L}$-$*$:1

We have $\alpha = \beta^{*:1}$ and $\overline{\alpha} = \overline{\beta}^*;?\mathrm{false}$ where $\beta \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\beta}$, and $[\![\langle\beta\rangle P]\!] = [\![\langle\overline{\beta}\rangle\overline{P}]\!]$ by the inductive hypothesis.

For the NDSd$\mathcal{L}$ formula, we have that $\omega \in [\![\langle \beta^{*:1} \rangle P]\!]$ iff $\exists (\mathrm{Det}(\omega), \nu) \in [\![\beta^{*:1}]\!]$, $\exists \pi \sim \nu$ where $\pi \neq \textbf{stuck}$, $\pi \in [\![P]\!]$.

But note that any $(\mathrm{Det}(\omega), \nu) \in [\![\beta^{*:1}]\!]$ must have $\nu = \mathrm{Det}(\textbf{stuck})$ by definition, so no such π exists. Then we have that $\omega \notin [\![\langle \beta^{*:1} \rangle P]\!]$ for all ω.

For the d$\mathcal{L}$ formula, note that no runs of $\overline{\beta}^*; ?\mathrm{false}$ exist, so we have that $\omega \notin [\![\langle \overline{\beta}^*; ?\mathrm{false} \rangle \overline{P}]\!]$ for all ω.

Thus, $\omega \in [\![\langle \beta^{*:1} \rangle P]\!]$ iff $\omega \in [\![\langle \overline{\beta}^*; ?\mathrm{false} \rangle \overline{P}]\!]$ as desired.

A.12.9 d$\mathcal{L}$-$*$:p

We have $\alpha = \beta^{*:p}$ and $\overline{\alpha} = \overline{\beta}^*$ where $0 < p < 1$, $\beta \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\beta}$, and $[\![\langle \beta \rangle P]\!] = [\![\langle \overline{\beta} \rangle \overline{P}]\!]$ by the inductive hypothesis.

For the NDSd$\mathcal{L}$ formula, we have that $\omega \in [\![\langle \beta^{*:p} \rangle P]\!]$ iff $\exists (\mathrm{Det}(\omega), \nu) \in [\![\beta^{*:p}]\!]$, $\exists \pi \sim \nu$ where $\pi \neq \textbf{stuck}$, $\pi \in [\![P]\!]$.

Now note that we must have $f_\nu = \sum_{i=0}^{\infty} (1-p) p^i f_{\xi_i} = f_{\xi_0}$ where $\xi_0 = \mathrm{Det}(\omega)$ for some $(\xi_i, \xi_{i+1}) \in [\![\beta]\!]$. Note also that all f_{ξ_i} have a non-zero coefficient since $0 < p < 1$.

Suppose there exists some $\pi \in \mathrm{supp}(\xi_k)$ for some ξ_k where $\pi \neq \textbf{stuck}$, $\pi \in [\![P]\!]$. Then we have that $\omega \in [\![\langle \beta^k \rangle P]\!]$, as we can cut off the chain of ξ_is at ξ_k. Using case d$\mathcal{L}$-;, we get that $\omega \in [\![\langle \overline{\beta}^k \rangle \overline{P}]\!]$. But note that $\exists k \in \mathbb{N}$, $\omega \in [\![\langle \overline{\beta}^k \rangle \overline{P}]\!]$ iff $\omega \in [\![\langle \overline{\beta}^* \rangle \overline{P}]\!]$.

Now suppose that no such π exists. Then we have that all $\tau \in \mathrm{supp}(f_\nu)$ is either **stuck**, or $\tau \notin [\![P]\!]$. Thus, $\omega \notin [\![\langle \beta^k \rangle P]\!]$ for all $k \in \mathbb{N}$, as otherwise some π would make a nonzero contribution to f_ν. But note that $\omega \notin [\![\langle \beta^k \rangle P]\!]$ for all $k \in \mathbb{N}$ iff $\omega \notin [\![\langle \overline{\beta}^* \rangle \overline{P}]\!]$.

In either case, $\omega \in [\![\langle \beta^{*:p} \rangle P]\!]$ iff $\omega in [\![\langle \overline{\beta}^* \rangle \overline{P}]\!]$.

A.12.10 d$\mathcal{L}$-$\cup$

We have $\alpha = \beta \cup \gamma$ and $\overline{\alpha} = \overline{\beta} \cup \overline{\gamma}$ where $\beta \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\beta}$, $\gamma \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\gamma}$, and $[\![\langle \beta \rangle P]\!] = [\![\langle \overline{\beta} \rangle \overline{P}]\!]$, $[\![\langle \gamma \rangle P]\!] = [\![\langle \overline{\gamma} \rangle \overline{P}]\!]$ by the inductive hypothesis.

$$
\begin{aligned}
[\![\langle \beta \cup \gamma \rangle P]\!] &= \{\omega : \exists (\mathrm{Det}(\omega), \nu) \in [\![\beta \cup \gamma]\!], \exists \pi \sim \nu \text{ where } \pi \neq \textbf{stuck}, \pi \in [\![P]\!]\} \\
&= \{\omega : \exists (\mathrm{Det}(\omega), \nu) \in [\![\beta]\!] \cup [\![\gamma]\!], \exists \pi \sim \nu \text{ where } \pi \neq \textbf{stuck}, \pi \in [\![P]\!]\} \\
&= \{\omega : \exists (\mathrm{Det}(\omega), \nu) \in [\![\beta]\!], \exists \pi \sim \nu \text{ where } \pi \neq \textbf{stuck}, \pi \in [\![P]\!]\} \\
&\quad \cup \{\omega : \exists (\mathrm{Det}(\omega), \nu) \in [\![\gamma]\!], \exists \pi \sim \nu \text{ where } \pi \neq \textbf{stuck}, \pi \in [\![P]\!]\} \\
&= [\![\langle \beta \rangle P]\!] \cup [\![\langle \gamma \rangle P]\!] \\
&= [\![\langle \overline{\beta} \rangle \overline{P}]\!] \cup [\![\langle \overline{\gamma} \rangle \overline{P}]\!] && \text{(inductive hypothesis)} \\
&= [\![\langle \overline{\beta} \cup \overline{\gamma} \rangle \overline{P}]\!] && (\mathsf{d}\mathcal{L} \text{ axiom})
\end{aligned}
$$

so $[\![\langle \beta \cup \gamma \rangle P]\!] = [\![\langle \overline{\beta} \cup \overline{\gamma} \rangle \overline{P}]\!]$ as desired.

A.12.11 $\mathsf{d}\mathcal{L}$-$\oplus$

We have $\alpha = \bigoplus_{i=1}^{n} p_i\beta_i$ and $\overline{\alpha} = \bigcup_{\substack{i=1\\p_i\neq 0}}^{n} \overline{\beta_i}$ where $\sum p_i = 1$, $\beta_i \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{\beta_i}$ and $[\![\langle\beta_i\rangle P]\!] = [\![\langle\overline{\beta_i}\rangle\overline{P}]\!]$ by the inductive hypothesis.

For the NDS$\mathsf{d}\mathcal{L}$ formula, we have that $\omega \in [\![\langle\bigoplus_{i=1}^{n} p_i\beta_i\rangle P]\!]$ iff $\exists(\mathrm{Det}(\omega),\nu) \in [\![\bigoplus_{i=1}^{n} p_i\beta_i]\!]$, $\exists \pi \sim \nu$ where $\pi \neq \textbf{stuck}$, $\pi \in [\![P]\!]$.

Now note that such a ν must have $f_\nu = \sum_{i=1}^{n} p_i f_{\xi_i}$ where $(\mathrm{Det}(\omega),\xi_i) \in [\![\beta_i]\!]$.

Suppose there exists some $\pi \in \mathrm{supp}(\xi_k)$ for some ξ_k where $p_k > 0$, $\pi \neq \textbf{stuck}$, $\pi \in [\![P]\!]$. Then we have that $\omega \in [\![\langle\beta_k\rangle P]\!]$. By the inductive hypothesis, that means that $\omega \in [\![\langle\overline{\beta_i}\rangle\overline{P}]\!]$. For the $\mathsf{d}\mathcal{L}$ formula, we have that $\omega \in [\![\langle \bigcup_{\substack{i=1\\p_i\neq 0}}^{n} \overline{\beta_i}\rangle\overline{P}]\!]$ since $p_k \neq 0$.

Now suppose that no such π exists. Then we have that all $\tau \in \mathrm{supp}(f_\nu)$ is either **stuck**, or $\tau \notin [\![P]\!]$. Thus, $\omega \notin [\![\langle\overline{\beta_i}\rangle\overline{P}]\!]$ for all β, as otherwise some π would make a nonzero contribution to f_ν. For the $\mathsf{d}\mathcal{L}$ formula, we would then have that $\omega \notin [\![\langle \bigcup_{\substack{i=1\\p_i\neq 0}}^{n} \overline{\beta_i}\rangle\overline{P}]\!]$.

In either case, $\omega \in [\![\langle\bigoplus_{i=1}^{n} p_i\beta_i\rangle P]\!]$ iff $\omega \in [\![\langle \bigcup_{\substack{i=1\\p_i\neq 0}}^{n} \overline{\beta_i}\rangle\overline{P}]\!]$ as desired.

A.12.12 $\mathsf{d}\mathcal{L}$-ODE

We have $\alpha = \{x' = f(x)\,\&\,Q\}$ and $\overline{\alpha} = \{x' = f(x)\,\&\,\overline{Q}\}$ where $Q \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} \overline{Q}$ and $[\![Q]\!] = [\![\overline{Q}]\!]$ by the inductive hypothesis.

Note that since $[\![Q]\!] = [\![\overline{Q}]\!]$, any ODE solution that satisfies $\{x' = f(x)\,\&\,\overline{Q}\}$ in $\mathsf{d}\mathcal{L}$ also satisfies $\{x' = f(x)\,\&\,Q\}$ in NDS$\mathsf{d}\mathcal{L}$, as the semantics for $\{x' = f(x)\,\&\,Q\}$ in NDS$\mathsf{d}\mathcal{L}$ matches the semantics for $\{x' = f(x)\,\&\,\overline{Q}\}$ in $\mathsf{d}\mathcal{L}$ except for using a NDS$\mathsf{d}\mathcal{L}$ formula instead of a $\mathsf{d}\mathcal{L}$ formula.

Note also that this case does not contain any probabilistic operators, so all distributions stay deterministic.

Thus, we must have $[\![\langle\{x' = f(x)\,\&\,Q\}\rangle P]\!] = [\![\langle\{x' = f(x)\,\&\,\overline{Q}\}\rangle\overline{P}]\!]$.

A.13 $\mathsf{d}\mathcal{L}$-$\langle\!|\,|\!\rangle$

We have $R = \langle\!|\alpha|\!\rangle P \leq p$ and $\overline{R} = Q$, and by the inductive hypothesis we have $\sigma(\rho(\alpha,P),p) \overset{\mathsf{d}\mathcal{L}}{\rightsquigarrow} Q$ where $\omega \in [\![\sigma(\rho(\alpha,P),p)]\!]$ iff $\omega \in [\![Q]\!]$.

Note that $[\![\langle\!|\alpha|\!\rangle P \leq p]\!] = [\![\sigma(\rho(\alpha,P),p)]\!]$, so $[\![\langle\!|\alpha|\!\rangle P \leq p]\!] = [\![Q]\!]$. Thus, $\omega \in [\![\langle\!|\alpha|\!\rangle P \leq p]\!]$ iff $\omega \in [\![Q]\!]$ as desired.

By induction we have that $\omega \in [\![R]\!]$ if $\omega \in [\![\overline{R}]\!]$.

B Proof of Theorem 2

Let $\omega \in [\![\langle\!|\alpha|\!\rangle P \leq p]\!]$. Then we will show that for all $(\mathrm{Det}(\omega), \nu) \in [\![\alpha]\!]$, $\mathbf{P}[\pi \in [\![P]\!]] \leq p$ for $\pi \sim \nu$.

Note that if $\omega \in [\![\langle\!|\alpha|\!\rangle P \leq p]\!]$, then $\omega \in [\![\sigma(\rho(\alpha, P), p)]\!]$. Thus, it must satisfy all preconditions generated by ρ, and we must have that $\omega[\![e]\!] \leq p$ where e is the final bound generated by ρ.

Now, let $\omega \in [\![\sigma(\rho(\alpha, P), p)]\!]$. We view α as a sequence of programs $\beta; \gamma$ where β can be empty and γ is not a sequential composition, and we induct on the last program γ as in the definition of ρ.

B.1 Empty case

Note that $\nu = \mathrm{Det}(\omega)$, since the program is empty. Then any such $\pi = \omega$.

Let e be the generated bound.

If $\omega \in [\![P]\!]$, then $\mathbf{P}[\pi \in [\![P]\!]] = 1$, so $\omega[\![e]\!] = 1$. If $\omega \notin [\![P]\!]$, then $\mathbf{P}[\pi \in [\![P]\!]] = 0$, so $\omega[\![e]\!] = 0$. This is exactly the precondition generated by ρ.

Thus, we have $\mathbf{P}[\pi \in [\![P]\!]] \leq p$ for all such $\pi = \omega$, since $\omega[\![e]\!] \leq p$.

B.2 $x := e$, $x := *$, $?Q$, α^*, $\alpha^{*:p}$, $\{x' = f(x)\ \&\ Q\}$

In all these cases, we have $\rho(\beta; \gamma, P) = \rho(\beta, \langle\gamma\rangle P)$.

If $\omega \in [\![\langle\!|\beta; \gamma|\!\rangle P \leq p]\!] = [\![\sigma(\rho(\beta; \gamma, P), p)]\!] = [\![\sigma(\rho(\beta, \langle\gamma\rangle P), p)]\!]$, then we have $\omega[\![e]\!] \leq p$ where ω satisfies all preconditions generated, and e is the bound generated from $\rho(\beta; \langle\gamma\rangle P)$.

Now consider $(\mathrm{Det}(\omega), \nu) \in [\![\beta; \gamma]\!]$. Then there exists some $(\mathrm{Det}(\omega), \xi) \in [\![\beta]\!], (\xi, \nu) \in [\![\gamma]\!]$, so by the inductive hypothesis we have $\mathbf{P}[\pi \in [\![\langle\gamma\rangle P]\!]] \leq p$ for $\pi \sim \xi$.

Consider $\pi \notin [\![\langle\gamma\rangle P]\!]$. Then note that all states that π maps to via γ cannot satisfy P, so the probability that states satisfy P after $\beta; \gamma$ is also bounded by p.

Thus, we also have for all $(\mathrm{Det}(\omega), \nu) \in [\![\beta; \gamma]\!]$, we have $\mathbf{P}[\pi \in [\![P]\!]] \leq p$ for $\pi \sim \xi$.

B.3 $x := \{p_1 : e_1, \ldots, p_n : e_n\}$

The semantics are identical to $\bigoplus_{i=1}^{n} p_i\{x := e_i\}$, and the definition of ρ is the same, so this case reduces to the $\oplus$ case.

B.4 $\alpha_1 \cup \alpha_2$

Then we have that $\omega \in [\![\langle\!|\beta; \alpha_1 \cup \alpha_2|\!\rangle P \leq p]\!]$.

By the inductive hypothesis, we have that if $\omega \in [\![\langle\!|\beta; \alpha_i|\!\rangle P \leq p]\!]$ then for all $(\mathrm{Det}(\omega), \nu) \in [\![\beta; \alpha_i]\!]$, $\mathbf{P}[\pi \in [\![P]\!]] \leq p$ for $\pi \sim \nu$.

If $\omega \in [\![\langle\!|\beta; \alpha_1 \cup \alpha_2|\!\rangle P \leq p]\!] = [\![\sigma(\rho(\beta; \alpha_1 \cup \alpha_2, P), p)]\!]$, then we have $\omega[\![\max\{e_1, e_2\}]\!] \leq p$ where ω satisfies all preconditions generated, and e_i is the bound generated from $\rho(\beta; \alpha_i, P)$.

But $\omega[\![\max\{e_1, e_2\}]\!] \leq p$ iff $\omega[\![e_1]\!] \leq p$ and $\omega[\![e_2]\!] \leq p$, so we must have that $\omega \in [\![\langle\!|\beta; \alpha_i|\!\rangle P \leq p]\!]$ for each α_i.

By the inductive hypothesis, we must then have for all $(\mathrm{Det}(\omega), \nu) \in [\![\beta; \alpha_i]\!]$, $\mathbf{P}[\pi \in [\![P]\!]] \leq p$ for $\pi \sim \nu$.

Note that $[\![\beta; \alpha_1 \cup \alpha_2]\!] = [\![\beta; \alpha_1]\!] \cup [\![\beta; \alpha_2]\!]$, so we have that for all $(\mathrm{Det}(\omega), \nu) \in [\![\beta; \alpha_1 \cup \alpha_2]\!]$, $\mathbf{P}[\pi \in [\![P]\!]] \leq p$ for $\pi \sim \nu$ as desired.

B.5 $\bigoplus_{i=1}^{n} p_i \alpha_i$

Then we have that $\omega \in [\![\langle\!| \beta; \bigoplus_{i=1}^{n} p_i \alpha_i |\!\rangle P \leq p]\!]$ where $\sum_{i=1}^{n} p_i = 1$.

By the inductive hypothesis, we have that if $\omega \in [\![\langle\!| \beta; \alpha_i |\!\rangle P \leq p]\!]$ then for all $(\mathrm{Det}(\omega), \nu) \in [\![\beta; \alpha_i]\!]$, $\mathbf{P}[\pi \in [\![P]\!]] \leq p$ for $\pi \sim \nu$.

If $\omega \in [\![\langle\!| \beta; \bigoplus_{i=1}^{n} p_i \alpha_i |\!\rangle P \leq p]\!] = [\![\sigma(\rho(\beta; \bigoplus_{i=1}^{n} p_i \alpha_i, P), p)]\!]$, then we have $\omega[\![\sum_{i=1}^{n} p_i e_i]\!] \leq p$ where ω satisfies all preconditions generated, and e_i is the bound generated from $\rho(\beta; \alpha_i, P)$.

Let $q_i = \omega[\![e_i]\!]$. Then we have that $\omega \in [\![\langle\!| \beta; \alpha_i |\!\rangle P \leq q_i]\!]$ for each α_i.

By the inductive hypothesis, we must then have for all $(\mathrm{Det}(\omega), \nu) \in [\![\beta; \alpha_i]\!]$, $\mathbf{P}[\pi \in [\![P]\!]] \leq q_i$ for $\pi \sim \nu$.

Thus, for all $(\mathrm{Det}(\omega), \nu) \in [\![\beta; \bigoplus_{i=1}^{n} p_i \alpha_i]\!]$, $\mathbf{P}[\pi \in [\![P]\!]] \leq \sum_{i=1}^{n} p_i q_i = p$ for $\pi \sim \nu$ as desired.

C Proof (sketch) of Theorem 3

Note that in the absence of probabilistic operators, the semantics for NDS$\mathtt{d}\mathcal{L}$ is the same as $\mathtt{d}\mathcal{L}$ if we identify each state with the corresponding deterministic distribution. (Note that for modalities, we ignore all state distributions that are fully stuck, so we have the same semantics for those.) Thus, any valid $\mathtt{d}\mathcal{L}$ formula is also a valid NDS$\mathtt{d}\mathcal{L}$ formula.

D Proof of Theorem 4

We will show that $[\![\alpha^*]\!] \subseteq [\![\{?\mathrm{true}\} \cup \{\alpha; \alpha^*\}]\!]$ and $[\![\alpha^*]\!] \supseteq [\![\{?\mathrm{true}\} \cup \{\alpha; \alpha^*\}]\!]$.

D.1 $\subseteq$

Suppose $(\mu, \nu) \in [\![\alpha^*]\!]$. We wish to show that $(\mu, \nu) \in [\![\{?\mathrm{true}\} \cup \{\alpha; \alpha^*\}]\!]$.

Note that $[\![\{?\mathrm{true}\} \cup \{\alpha; \alpha^*\}]\!] = [\![\{?\mathrm{true}\}]\!] \cup [\![\{\alpha; \alpha^*\}]\!]$.

Note also that there exist $n \in \mathbb{N}$ and ξ_i such that $\xi_0 = \mu$, $\xi_n = \nu$, and $(\xi_i, \xi_{i+1}) \in [\![\alpha]\!]$ $\forall\, 0 \leq i \leq n-1$.

If $n = 0$, then we have $\mu = \nu$. Note that $[\![\{?\mathrm{true}\}]\!] = \{(\mu, \mu) : \mu \in \mathbf{SP}\}$, so $(\mu, \nu) \in [\![\{?\mathrm{true}\}]\!]$, so $(\mu, \nu) \in [\![\{?\mathrm{true}\} \cup \{\alpha; \alpha^*\}]\!]$ as desired.

If $n > 0$, then we have $(\xi_1, \nu) \in [\![\alpha^*]\!]$. Note that $(\mu, \xi_1) \in [\![\alpha]\!]$, so $(\mu, \nu) \in [\![\{\alpha; \alpha^*\}]\!]$, so $(\mu, \nu) \in [\![\{?\mathrm{true}\} \cup \{\alpha; \alpha^*\}]\!]$ as desired.

D.2 $\supseteq$

Suppose $(\mu, \nu) \in [\![\{?\mathrm{true}\} \cup \{\alpha; \alpha^*\}]\!]$. We wish to show that $(\mu, \nu) \in [\![\alpha^*]\!]$.

Note that $[\![\{?\mathrm{true}\} \cup \{\alpha; \alpha^*\}]\!] = [\![\{?\mathrm{true}\}]\!] \cup [\![\{\alpha; \alpha^*\}]\!]$, so either $(\mu, \nu) \in [\![\{?\mathrm{true}\}]\!]$ or $(\mu, \nu) \in [\![\{\alpha; \alpha^*\}]\!]$.

If $(\mu, \nu) \in [\![\{?\mathrm{true}\}]\!]$, then we have $\mu = \nu$. Then we can let $\nu = \xi_0 = \mu$, so $(\mu, \nu) \in [\![\alpha^*]\!]$ as desired.

If $(\mu,\nu) \in [\![\{\alpha;\alpha^*\}]\!]$, then there exists ζ such that $(\mu,\zeta) \in [\![\alpha]\!]$ and $(\zeta,\nu) \in [\![\alpha^*]\!]$. Then there exist $n \in \mathbb{N}$ and ξ_i such that $\xi_0 = \zeta$, $\xi_n = \nu$, and $(\xi_i,\xi_{i+1}) \in [\![\alpha]\!]$ $\forall\, 0 \leq i \leq n-1$. If we let $\zeta_0 = \mu$, $\zeta_i = \xi_{i-1}$, and use $n+1$ in the definition of $[\![\alpha^*]\!]$, then we get that $(\mu,\nu) \in [\![\alpha^*]\!]$ as desired.

E Proof of Theorem 5

We will show that $[\![\alpha^{*:p}]\!] \subseteq [\![(1-p)\{?\text{true}\}\oplus p\{\alpha;\alpha^{*:p}\}]\!]$ and $[\![\alpha^{*:p}]\!] \supseteq [\![(1-p)\{?\text{true}\}\oplus p\{\alpha;\alpha^{*:p}\}]\!]$.

E.1 $\subseteq$

Suppose $(\mu,\nu) \in [\![\alpha^{*:p}]\!]$. We wish to show that $(\mu,\nu) \in [\![(1-p)\{?\text{true}\} \oplus p\{\alpha;\alpha^{*:p}\}]\!]$.

Then we have that $f_\nu = \sum_{i=0}^{\infty}(1-p)p^i f_{\xi_i}$ for some $(\xi_i,\xi_{i+1}) \in [\![\alpha]\!]$ where $\xi_0 = \mu$.

Then $f_\nu = (1-p)f_{\xi_0} + p\sum_{i=0}^{\infty}(1-p)p^i f_{\xi_{i+1}}$ for some $(\xi_i,\xi_{i+1}) \in [\![\alpha]\!]$ where $\xi_0 = \mu$.

Let $\zeta_i = \xi_{i+1}$ and define ζ with pmf $f_\zeta = \sum_{i=0}^{\infty}(1-p)p^i f_{\zeta_i}$. Then $f_\nu = (1-p)f_\mu + pf_\zeta$.

Now note that $(\zeta_0,\zeta) \in [\![\alpha^{*:p}]\!]$ and $(\mu,\zeta_0) \in [\![\alpha]\!]$, so $(\mu,\zeta) \in [\![\alpha;\alpha^{*:p}]\!]$.

Since $[\![\{?\text{true}\}]\!] = \{(\mu,\mu) : \mu \in \mathbf{SP}\}$, we then have $(\mu,\nu) \in [\![(1-p)\{?\text{true}\} \oplus p\{\alpha;\alpha^{*:p}\}]\!]$ as desired.

E.2 $\supseteq$

Suppose $(\mu,\nu) \in [\![(1-p)\{?\text{true}\} \oplus p\{\alpha;\alpha^{*:p}\}]\!]$. We wish to show that $(\mu,\nu) \in [\![\alpha^{*:p}]\!]$.

Then we have that $f_\nu = (1-p)f_{\xi_0} + f_\zeta$ for some $(\mu,\xi_0) \in [\![?\text{true}]\!]$ and $(\mu,\zeta) \in [\![\alpha;\alpha^{*:p}]\!]$.

Note that then $\xi_0 = \mu$.

Then $(\mu,\zeta_0) \in [\![\alpha]\!]$ and $(\zeta_0,\zeta) \in [\![\alpha^{*:p}]\!]$ for some ζ_0.

Then $f_\zeta = \sum_{i=0}^{\infty}(1-p)p^i f_{\zeta_i}$ for $(\zeta_i,\zeta_{i+1}) \in [\![\alpha]\!]$.

Let $\xi_i = \zeta_{i-1}$ for $i \geq 1$.

Then we have $f_\nu = (1-p)f_{\xi_0} + f_\zeta = f_\nu = (1-p)f_{\xi_0} + \sum_{i=0}^{\infty}(1-p)p^i f_{\zeta_i} = \sum_{i=0}^{\infty}(1-p)p^i f_{\xi_i}$.

Since $\xi_0 = \mu$ and $(\xi_i,\xi_{i+1}) \in [\![\alpha]\!]$, we then have $(\mu,\nu) \in [\![\alpha^{*:p}]\!]$ as desired.