Decidability Results for Probabilistic Hybrid Automata

Prof. Dr. Erika Ábrahám

Informatik 2 - Theory of Hybrid Systems
RWTH Aachen

SS09

Literatur

Jeremy Sproston:
Decidable Model Checking of Probabilistic Hybrid Automata
FTRTFT'00, LNCS 1926, pp. 31–45, 2000.

Motivation

Which components of a hybrid system could be probabilistic?

Motivation

Which components of a hybrid system could be probabilistic?

Example applications?

Motivation

Which components of a hybrid system could be probabilistic?

Example applications?

What do you expect to be a decidable class?

Definition (Distribution)

- For a set Y, a (discrete probability) distribution on Y is a function $\mu : Y \to [0, 1]$ such that $\mu(y) > 0$ for at most countably many $y \in Y$ and $\sum_{y \in Y} \mu(y) = 1$.

Definition (Distribution)

- For a set Y, a (discrete probability) distribution on Y is a function $\mu : Y \to [0, 1]$ such that $\mu(y) > 0$ for at most countably many $y \in Y$ and $\sum_{y \in Y} \mu(y) = 1$.
- We use $\mathrm{Dist}(Y)$ to denote the set of all distributions on Y.

Definition (Distribution)

- For a set Y, a (discrete probability) distribution on Y is a function $\mu : Y \to [0, 1]$ such that $\mu(y) > 0$ for at most countably many $y \in Y$ and $\sum_{y \in Y} \mu(y) = 1$.
- We use $\text{Dist}(Y)$ to denote the set of all distributions on Y.
- For a distribution μ on a set Y let $\text{support}(\mu)$ be the set of elements y of Y with $\mu(y) > 0$.

Questions

How can probabilities and conditional jumps with effects be combined?

Probabilistic hybrid automata

(We skip synchronization labels.)

Definition (Probabilistic hybrid automaton)

A probabilistic hybrid automaton H is a hybrid automaton without the Edge set, and with

- an additional function prob which maps to each location a finite set of probability distributions on $\text{Loc} \times 2^{\mathbb{R}^n} \times 2^{\text{Var}}$, and

Probabilistic hybrid automata

(We skip synchronization labels.)

Definition (Probabilistic hybrid automaton)

A probabilistic hybrid automaton H is a hybrid automaton without the Edge set, and with

- an additional function prob which maps to each location a finite set of probability distributions on $\text{Loc} \times 2^{\mathbb{R}^n} \times 2^{\text{Var}}$, and
- a function pre which maps to each location l and each distribution in $\text{prob}(l)$ a subset of $\mathbb{R}^n$, called the precondition set.

Probabilistic hybrid automata

(We skip synchronization labels.)

Definition (Probabilistic hybrid automaton)

A probabilistic hybrid automaton H is a hybrid automaton without the Edge set, and with

- an additional function prob which maps to each location a finite set of probability distributions on $\text{Loc} \times 2^{\mathbb{R}^n} \times 2^{\text{Var}}$, and
- a function pre which maps to each location l and each distribution in $\text{prob}(l)$ a subset of $\mathbb{R}^n$, called the precondition set.

Probabilistic hybrid automata

(We skip synchronization labels.)

Definition (Probabilistic hybrid automaton)

A probabilistic hybrid automaton H is a hybrid automaton without the Edge set, and with

- an additional function prob which maps to each location a finite set of probability distributions on $\text{Loc} \times 2^{\mathbb{R}^n} \times 2^{\text{Var}}$, and
- a function pre which maps to each location l and each distribution in $\text{prob}(l)$ a subset of $\mathbb{R}^n$, called the precondition set.

Definition (Probabilistic rectangular automaton)

A probabilistic rectangular automaton is a probabilistic hybrid automaton with only rectangular sets in the definition.

Semantics

- Flows as before.

Semantics

- Flows as before.
- A jump can take place from a source state (l, ν) to a target state (l', ν') iff

Semantics

- Flows as before.
- A jump can take place from a source state (l, ν) to a target state (l', ν') iff
  - there is a distribution $\mu \in \text{prob}(l)$ such that

Semantics

- Flows as before.
- A jump can take place from a source state (l, ν) to a target state (l', ν') iff
  - there is a distribution $\mu \in \text{prob}(l)$ such that
  - the precondition $\text{pre}(l)(\mu)$ is satisfied by ν, and

Semantics

- Flows as before.
- A jump can take place from a source state (l, ν) to a target state (l', ν') iff
  - there is a distribution $\mu \in \text{prob}(l)$ such that
  - the precondition $\text{pre}(l)(\mu)$ is satisfied by ν, and
  - $\mu((l', post, X)) > 0$ for some $post \subseteq \mathbb{R}^n$ and $X \subseteq \text{Var}$ with

Semantics

- Flows as before.
- A jump can take place from a source state (l, ν) to a target state (l', ν') iff
  - there is a distribution $\mu \in \text{prob}(l)$ such that
  - the precondition $\text{pre}(l)(\mu)$ is satisfied by ν, and
  - $\mu((l', post, X)) > 0$ for some $post \subseteq \mathbb{R}^n$ and $X \subseteq \text{Var}$ with
  - $\nu' \in post$ and

Semantics

- Flows as before.
- A jump can take place from a source state (l, ν) to a target state (l', ν') iff
  - there is a distribution $\mu \in \text{prob}(l)$ such that
  - the precondition $\text{pre}(l)(\mu)$ is satisfied by ν, and
  - $\mu((l', post, X)) > 0$ for some $post \subseteq \mathbb{R}^n$ and $X \subseteq \text{Var}$ with
  - $\nu' \in post$ and
  - $\nu(x) = \nu'(x)$ for all $x \in \text{Var} \backslash X$.

Questions

What is the maximal probability of a single path?

Questions

What is the maximal probability of a single path?

What is about time divergence?

Questions

What is the maximal probability of a single path?

What is about time divergence?

What is about zeno behaviour?

Adversaries

Intuitively, an adversary resolves all of the nondeterministic choices of a probabilistic hybrid automaton.

Adversaries

Intuitively, an adversary resolves all of the nondeterministic choices of a probabilistic hybrid automaton.

Definition (Adversary)

An adversary of a probabilistic hybrid automaton H is a function A mapping each finite path ω with last state (l, ν) of H to a distribution $\mu \in \text{prob}(l)$.

Adversaries

Intuitively, an adversary resolves all of the nondeterministic choices of a probabilistic hybrid automaton.

Definition (Adversary)

An adversary of a probabilistic hybrid automaton H is a function A mapping each finite path ω with last state (l, ν) of H to a distribution $\mu \in \text{prob}(l)$.

Definition

An adversary A of a probabilistic hybrid automaton H is divergent iff for each state of H the total probability of the divergent paths under A is 1. Let $\mathcal{A}_H$ be the set of divergent adversaries of H.

Adversaries

Intuitively, an adversary resolves all of the nondeterministic choices of a probabilistic hybrid automaton.

Definition (Adversary)

An adversary of a probabilistic hybrid automaton H is a function A mapping each finite path ω with last state (l, ν) of H to a distribution $\mu \in \text{prob}(l)$.

Definition

An adversary A of a probabilistic hybrid automaton H is divergent iff for each state of H the total probability of the divergent paths under A is 1. Let $\mathcal{A}_H$ be the set of divergent adversaries of H.

Definition

A probabilistic hybrid automaton is non-zeno iff it has at least one divergent adversary.

Question

How could a logic

arguing about timed and probabilistic behaviour

look like?

Probabilistic Timed Computation Tree Logic

Definition (PTCTL Syntax)

The abstract syntax of PTCTL is as follows:

$$\Phi \ ::= a \mid g \mid \Phi \wedge \Phi \mid \neg\Phi \mid z.\Phi \mid P_{\sim\lambda}[\Phi \mathcal{U} \Phi]$$

with a an atomic proposition, g a clock constraint, z a formula clock, $\sim \in \{\leq, <, >, \geq\}$, and $\lambda \in [0, 1]$.

PTCTL Semantics

Definition

$$\sigma, \mathcal{E} \models z.\Phi \quad \leftrightarrow \quad \sigma, \mathcal{E}[z := 0] \models \Phi$$

PTCTL Semantics

Definition

$$\sigma, \mathcal{E} \models z.\Phi \quad \leftrightarrow \quad \sigma, \mathcal{E}[z := 0] \models \Phi$$

$$\sigma, \mathcal{E} \models P_{\sim\lambda}[\Phi_1 \mathcal{U} \Phi_2] \quad \leftrightarrow \quad \textit{for all divergent adversaries } A \in \mathcal{A}_H, \textit{ the total probability of all infinite pathes } \omega \textit{ under } A \textit{ with } \omega, \mathcal{E} \models \Phi_1 \mathcal{U} \Phi_2 \textit{ is } \sim \lambda.$$

PTCTL Semantics

Definition

$$
\begin{array}{lcl}
\sigma, \mathcal{E} \models z.\Phi & \leftrightarrow & \sigma, \mathcal{E}[z := 0] \models \Phi \\
\sigma, \mathcal{E} \models P_{\sim\lambda}[\Phi_1 \mathcal{U} \Phi_2] & \leftrightarrow & \textit{for all divergent adversaries } A \in \mathcal{A}_H, \\
 & & \textit{the total probability of all infinite pathes } \omega \\
 & & \textit{under } A \textit{ with } \omega, \mathcal{E} \models \Phi_1 \mathcal{U} \Phi_2 \textit{ is } \sim \lambda.
\end{array}
$$

Remember: $\Phi_1 \mathcal{U} \Phi_2$ in TCTL corresponds to $(\Phi_1 \vee \Phi_2) \mathcal{U} \Phi_2$ in CTL.

Decidability results

Lemma

The PTCTL model checking problem for initialized probabilistic rectangular automata is decidable.

Decidability results

Lemma

The PTCTL model checking problem for initialized probabilistic rectangular automata is decidable.

Probabilistic timed automaton

↑

Probabilistic initialized stopwatch automaton

↑

Probabilistic initialized singular automaton

↑

Probabilistic initialized rectangular automaton

A probabilistic timed automaton is a probabilistic rectangular automaton with deterministic jumps such that every variable is a clock, i.e., $Act(l)(x) = [1, 1]$ for all locations l and variables x.

A probabilistic timed automaton is a probabilistic rectangular automaton with deterministic jumps such that every variable is a clock, i.e., $Act(l)(x) = [1, 1]$ for all locations l and variables x.

Lemma

The PTCTL model checking problem for probabilistic timed automata is decidable.

A probabilistic timed automaton is a probabilistic rectangular automaton with deterministic jumps such that every variable is a clock, i.e., $Act(l)(x) = [1, 1]$ for all locations l and variables x.

Lemma

The PTCTL model checking problem for probabilistic timed automata is decidable.

Model checking as for timed automata with

- summing up probabilities for distibutions, and
- taking minimum/maximum of all distributions within a location.

Decidability results

A probabilistic stopwatch automaton is a probabilistic rectangular automaton with deterministic jumps and stopwatch variables only.

Decidability results

A probabilistic stopwatch automaton is a probabilistic rectangular automaton with deterministic jumps and stopwatch variables only.

Probabilistic timed automaton

↑

Probabilistic initialized stopwatch automaton

Decidability results

A probabilistic stopwatch automaton is a probabilistic rectangular automaton with deterministic jumps and stopwatch variables only.

Probabilistic timed automaton

$\uparrow$

Probabilistic initialized stopwatch automaton

Construction is similar as for non-probabilistic automata (probabilistic setting: adapt preconditions).

Decidability results

A probabilistic singular automaton is a probabilistic rectangular automaton with deterministic jumps such that every variable of the automaton is a finite-slope variable.

Decidability results

A probabilistic singular automaton is a probabilistic rectangular automaton with deterministic jumps such that every variable of the automaton is a finite-slope variable.

Probabilistic imed automaton

↑

Probabilistic initialized stopwatch automaton

↑

Probabilistic initialized singular automaton

Decidability results

A probabilistic singular automaton is a probabilistic rectangular automaton with deterministic jumps such that every variable of the automaton is a finite-slope variable.

Probabilistic imed automaton

↑

Probabilistic initialized stopwatch automaton

↑

Probabilistic initialized singular automaton

Construction is similar as for non-probabilistic automata (probabilistic setting: adapt pre- and postconditions).

Decidability results

Probabilistic timed automaton

↑

Probabilistic initialized stopwatch automaton

↑

Probabilistic initialized singular automaton

↑

Probabilistic initialized rectangular automaton

Decidability results

Probabilistic timed automaton

↑

Probabilistic initialized stopwatch automaton

↑

Probabilistic initialized singular automaton

↑

Probabilistic initialized rectangular automaton

Construction is similar as for non-probabilistic automata (probabilistic setting: adapt all conditions, copies of distributions).