Probabilistic Program Analysis with Martingales

Paper by Chakarov and Sankaranarayanan

Frederik Zwilling

Seminar Probabilistic Programs, RWTH Aachen University

Agenda

Teaser

Martingales

- Similar to expectation invariants
- Discrete stochastic process $\{M_n\}$
- Expectation in the next step equals the current value
  $\mathbb{E}(M_n \mid m_{n-1}, ..., m_0) = m_{n-1}$

Use in Probabilistic Program Analysis

- Derive probabilistic assertions (e.g. $Pr(x \in [200, 300])$)
- Prove almost sure termination
- Discovering martingales automatically

Motivation

Probabilistic Programs

- Useful, getting more widespread
- Verification important

Why Martingales for Analysis?

- Well known in mathematics
- Usable theorems to bound differences in experiments

Automated Verification

- Practical programs too large for manual verification

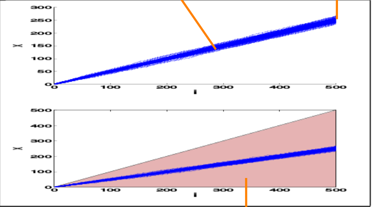
Motivating Example: Assertion

Computes sum of 500 random samples

$\mathbb{E}(x_{500}) = 250$

$\sim$ 50 sample execution paths

```
real x = 0;
real N = 500;
for ( i=0; i < N; ++i )
  x = x + unifRand(0,1);
// Pr(x∈[200,300]) ?
```

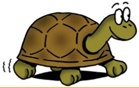


Probabilistic Assertion

Area of possible execution paths

Motivating Example: Termination

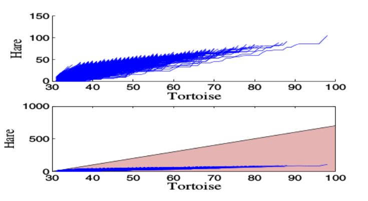
Race Hare vs. Tortoise



$\mathbb{E}(\Delta h) = 2.5$

$\mathbb{E}(\Delta t) = 1$

```
real h, t;
// h is hare and t is tortoise
h = 0; t = 30;
while ( h <= t ){
  if (flip (0.5) )
       h = h + unifRand(0,10);
  t = t +1;
} // almost sure terminate?
```



Terminates when hare surpasses tortoise

$\Rightarrow$ Program should terminate after some time

Agenda

Probabilistic Transition System (PTS)

```
real h, t;
// h is hare and t is tortoise
h = 0; t = 30;
while ( h <= t ){
  if (flip (0.5) )
      h = h + unifRand(0,10);
  t = t +1;
} // almost sure terminate?
```

$\Rightarrow$ l_4 l_f

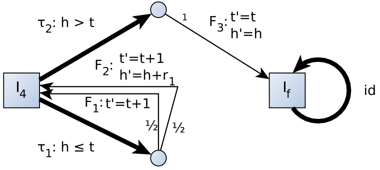
A PTS Π is a tuple $\langle X, R, L, \mathcal{T}, l_0, x_0, l_f \rangle$

- X / R: vectors of program variables/random variables
- L: finite set of locations
- l_0 / x_0: initial location/program variable values
- l_f: final location (terminated)

Probabilistic Transition System (PTS)

```
real h, t;
// h is hare and t is tortoise
h = 0; t = 30;
while ( h <= t ){
  if (flip (0.5) )
        h = h + unifRand(0,10);
  t = t +1;
} // almost sure terminate?
```

$\Rightarrow$



A PTS Π is a tuple $\langle X, R, L, \mathcal{T}, l_0, x_0, l_f \rangle$

- $\mathcal{T}$: finite set of transitions
  $\tau \in \mathcal{T}$ is a tuple $\langle l, \phi, f_1, \ldots, f_k \rangle$ with
  - Source location $l \in L$, guard assertion ϕ over X,
  - Forks $f_1, \ldots, f_k$ where
    f_i is a tuple (p_i, F_i, m_i) with
    - fork probability p_i
    - destination location $m_i \in L$
    - update function $F_i(X, R)$

PTS continued

State of a PTS

Tuple $s = (l, x)$, where

- Location $l \in L$
- x variable valuation of X

No Demonic Restriction

- Transition choice deterministic
- Exactly one transition $\tau(s)$ enabled for every state s

Sample Executions and Almost Sure Termination

Sample Execution σ

- Valid sequence of states $(l_0, x_0) \xrightarrow{\tau_1} (l_1, x_1) \xrightarrow{\tau_2} \dots$
- *Terminating* if it reaches state (l_f, x) for some x
- Syntactic Path π of σ $l_0 \xrightarrow{\tau_1} l_1 \xrightarrow{\tau_2} \dots$
- Probability $\mu(\pi) \in [0, 1]$: product of all used fork probabilities

Example from  :

$(l_4, (h = 0, t = 30)) \xrightarrow{\tau_1}$
$(l_4, (h = 0, t = 31)) \xrightarrow{\tau_1}$
$(l_4, (h = 8, t = 32)) \xrightarrow{\tau_1}$
...

Almost-Sure Termination

PTS *terminates almost surely* iff $\sum_{\pi \text{ terminating}} \mu(\pi) = 1$

Post-Expectation

Post-Distribution `Post-Distrib`(s)

- Distribution of states after the next execution step
- Depends on forks of $\tau(s)$ and distribution of R

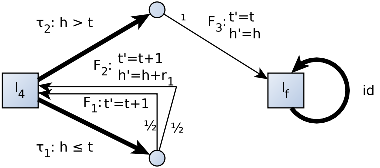
Post-Expectation $\mathbb{E}_\tau(e \mid s)$

Expected value of expression e over `Post-Distrib`(s)

$$\mathbb{E}_{\tau(s)}(e \mid s) := \sum_{i=1}^{k} p_i * \mathbb{E}_R(e[x/F_i(x, r)]) \text{ , where}$$

- Transition: $\tau(s) = \langle l, \phi, f_1, ..., f_k \rangle$
- Fork: $f_i = (p_i, F_i, m_i)$

Example Post-Expectation



- Current state $s = (l_4, (h, t))$, $h \leq t$
- Expression $e = 5t - 2h$
- Random sample $r_1 \in \mathit{unifRand}(0, 10)$

$$\begin{aligned}\mathbb{E}_{\tau_1}(e \mid s) &= \frac{1}{2}\mathbb{E}(e[x/F_1(h,t,r_1)]) + \frac{1}{2}\mathbb{E}(e[x/F_2(h,t,r_1)]) \\ &= \frac{1}{2}\mathbb{E}(5(t+1) - 2h) + \frac{1}{2}\mathbb{E}(5(t+1) - 2(h + r_1)) \\ &= 5t - 2h + 5 - \mathbb{E}(r_1) = 5t - 2h = e\end{aligned}$$

Martingale and Super Martingale Expressions

Martingale Expression

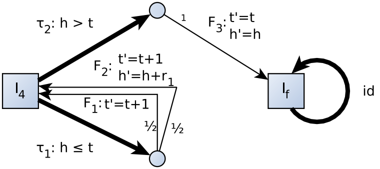
- Expression e over X for a PTS Π
- For every state $s = (l, x)$ the post-expectation of e equals the current value of e under x

$$\forall s = (l, x) : \mathbb{E}_{\tau(s)}(e \mid s) = e$$

Likewise, e is a Super Martingale iff

$$\forall s = (l, x) : \mathbb{E}_{\tau(s)}(e \mid s) \leq e$$

Example Super Martingale



- Current state $s = (l_4, (h, t))$, $h \leq t$
- Random sample $r_1 \in unifRand(0, 10)$
- Expression $e = t - h$

$$\mathbb{E}_{\tau_1}(e \mid s) = \frac{1}{2}\mathbb{E}(t + 1 - h) + \frac{1}{2}\mathbb{E}(t + 1 - (h + r_1))$$
$$= t - h + 1 - \frac{1}{2}\mathbb{E}(r_1) = t - h - 1.5 \leq e$$

- For states (l_f, x) and $(l_4, (h, t))$ with $h > t$ trivial

$\Rightarrow$ $t - h$ is a super martingale

Agenda

Azuma-Hoeffding Theorem

Azuma-Hoeffding Theorem

- $\{M_n\}$ super martingale
- $|m_n - m_{n-1}| < c$, for constant c

$\Rightarrow$ $\forall n \in \mathbb{N}$ and $\forall t \in \mathbb{R}_0^+$:

$$Pr(M_n - M_0 \geq t) \leq exp\left(\frac{-t^2}{2nc^2}\right)$$

- For martingale $\{M_n\}$:

$$Pr(|M_n - M_0| \geq t) \leq 2 * exp\left(\frac{-t^2}{2nc^2}\right)$$

- For us: $M_n = (e)_n$ value in n-th execution step

Example: Probabilistic Assertion

```
real x = 0;
real N = 500;
for ( i=0; i < N; ++i )
  x = x + unifRand(0,1);
// Pr(x∈[200,300]) ?
```

Find martingale by intuition

- x expected to be increased by 0.5 each iteration
- i incremented by 1 each iteration

$\Rightarrow$ Martingale $e = 2x - i$

Find parameters t and c

$$Pr(|M_n - M_0| \geq t) = Pr(|(2x - i)_{500} - (2x - i)_0| \geq t)$$

$$= Pr(|2x - 500 - 0| \geq t) = Pr(|x - 250| \geq \frac{t}{2})$$

$$= 1 - Pr(|x - 250| < \frac{t}{2})$$

- $t = 100$ yields $Pr(x \in [200, 300])$
- $max(e_{n+1} - e_n) = 1 \Rightarrow c = 1$

Example: Probabilistic Assertion

What we have

- $n = 500$, $c = 1$, $t = 100$
- $Pr(|M_n - M_0| \geq t) = 1 - Pr(|x - 250| < \frac{t}{2})$

Use Azuma-Hoeffding Theorem

$$Pr(|M_n - M_0| \geq t) \leq 2 * exp\left(\frac{-t^2}{2nc^2}\right)$$

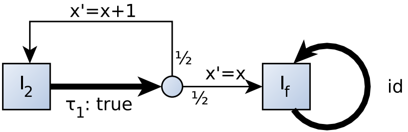
$$\begin{aligned} Pr(x \in [200, 300]) &\geq Pr(x \in (200, 300)) \\ &= 1 - Pr(|M_{500} - M_0| \geq 100) \\ &\geq 1 - 2exp\left(\frac{-100^2}{2 * 500 * 1}\right) \geq 0.999909 \end{aligned}$$

Agenda

Martingale Expression Maps

- Expression not enough to reason about termination

```
int x := 0;
while (flip (0.5))
   x ++;
// end
```



- Use expression map η instead:

$$\eta(l) = \begin{cases} x & , l = l_2 \\ x - 1 & , l = l_f \end{cases}$$

$$\begin{aligned} \mathbb{E}_{\tau_1}(\eta \mid (l_2, x)) &= \frac{1}{2}\eta(l_2)[x/x+1] + \frac{1}{2}\eta(l_f)[x/x] \\ &= \frac{1}{2}(x+1) + \frac{1}{2}(x-1) = x = \eta(l_2) \end{aligned}$$

Martingale Expression Maps

- Post-Expectation for expression map:
  $\mathbb{E}_\tau(\eta \mid s) = \sum_{i=1}^{k} p_i * \mathbb{E}(\eta(l_i)[x/F_i(x,r)])$

Martingale Expression Map

- Generalized version of the Martingale Expression
- Expression map η over X for a PTS Π
- For every state $s = (l, x)$ the post-expectation of η equals the current value of $\eta(l)$ under x

$$\forall s = (l, x) : \mathbb{E}_{\tau(s)}(\eta \mid s) = \eta(l)$$

- Similarly for super martingale expression maps

$$\forall s = (l, x) : \mathbb{E}_{\tau(s)}(\eta \mid s) \leq \eta(l)$$

Almost-Sure Termination - Intuitive Approach


```
real h, t;
// h is hare and t is tortoise
h = 0; t = 30;
while ( h <= t ){
  if (flip (0.5) )
      h = h + unifRand(0,10);
  t = t +1;
} // almost sure terminate?
```

Super martingale $t - h$

- Distance between hare and tortoise
- Loop guard $t - h \geq 0$

$\Rightarrow$ Indicates termination

Idea to prove almost-sure termination

- Find super martingale expression map η with $\eta(s) < 0$ iff s is in I_f
- Show convergence to value smaller than 0

Super Martingale Ranking Function

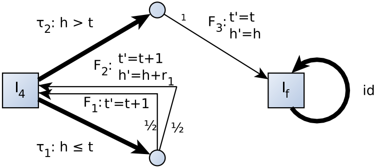
Super Martingale Ranking Function (SMRF) η

- Super martingale expression map of a PTS Π with
- $\eta(l) \geq 0$ for all $l \in L \setminus \{l_f\}$
- $\eta(l_f) \in [-K, 0)$ for some bound K
- For some constant $\epsilon > 0$ and all transitions $\tau \in \mathcal{T} \setminus \{id\}$:
  $\forall s = (l, x) : \mathbb{E}_{\tau(s)}(\eta \mid s) \leq \eta(l) - \epsilon$

Almost-Sure Termination

The PTS Π has a SMRF $\eta \Rightarrow \Pi$ terminates almost surely (sound, but not complete)

Example: Almost-Sure Termination



Construction of SMRF

- Super martingale $t - h$
- $t - h + 9$ also super martingale
- $\eta(l) = \begin{cases} t - h + 9 & , l = l_4 \\ t - h & , l = l_f \end{cases}$
- '+9' ensures $\eta(l_4) \geq 0$

- Program reaches $l_f \Rightarrow \eta(l_f) < 0$
- Choose $\epsilon = 1$
  - For τ_1 (loop iteration):
    $\mathbb{E}_{\tau_1}(\eta \mid s) = t - h + 9 - 1.5 \leq \eta(l_4) - \epsilon$
  - For τ_2 (loop exit):
    $\mathbb{E}_{\tau_2}(\eta \mid s) = \eta(l_f) = t - h \leq t - h + 9 - 1 = \eta(l_4) - \epsilon$

$\Rightarrow$ η is a SMRF $\Rightarrow$ almost-sure termination

Agenda

Martingale Synthesis

How to find martingales automatically?

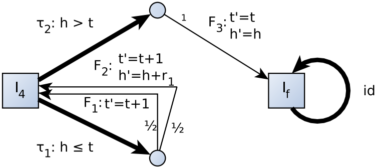
- Affine PTS (only linear transition guards and linear update functions $F_i(x, r) = A_i x + B_i r + a_i$)
- Linear martingale expression $e = c^T x$
- Linear restrictions for martingale property $\forall x : \phi_{\tau(s)}(x) \Rightarrow \mathbb{E}_{\tau(s)}(e \mid s) = e$

How to find SMRF?

- More variables (ϵ, c_l for all $l \in L$)
- Properties of SMRF as linear constraints $\epsilon > 0$, $\eta(l_f) < 0$ and $\eta(l) \geq 0$ for all $l \in L \backslash \{l_f\}$
- Linear restrictions for martingale property $\forall x : \phi_{\tau(s)}(x) \Rightarrow \mathbb{E}_{\tau(s)}(\eta \mid s) \leq \eta - \epsilon$

$\Rightarrow$ Solve with linear algebra

Martingale Synthesis Example



- Goal: find martingale
  $$e = c^T x = \begin{pmatrix} c_1 & c_2 \end{pmatrix} \begin{pmatrix} h \\ t \end{pmatrix}$$
- Constraint from τ_2 trivial

- Constraint for state $s = (l_4, (h, t))$ with $h \leq t$:

$$\mathbb{E}_{\tau_1}(e \mid s) = \frac{1}{2}\mathbb{E}(c^T \begin{pmatrix} h \\ t+1 \end{pmatrix}) + \frac{1}{2}\mathbb{E}(c^T \begin{pmatrix} h + r_1 \\ t+1 \end{pmatrix}) = c_1 h + c_2 t$$

$$\Leftrightarrow c_1 h + c_2 t + c_2 + \frac{1}{2} c_1 \mathbb{E}(r_1) = c_1 h + c_2 t$$

$$\overset{\mathbb{E}(r_1)=5}{\Leftrightarrow} c_2 + \frac{5}{2} c_1 = 0$$

$\Rightarrow$ e.g. $5t - 2h$ is a martingale

Martingale Synthesis

Evaluation

- Important step towards automated analysis
- Implementation provided by authors
- Still much manual effort needed
  (e.g. find parameters for Azuma-Hoeffding)
- Only for linear programs

Agenda

Related Work

Similar work: Quantitative Invariants [McIver, MCS'06]

- Similar notation to super martingales
- Proposed by McIver and Morgan
- Almost-sure termination proof
- Allows demonic non-determinism

Generalization of approach [Chakarov, SA'14]

- Later work by Chakarov and Sankaranarayanan
- Analysis of probabilistic program loops

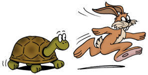
Conclusion

Probabilistic Program Analysis with Martingales

- Find martingales (automatically for linear ones)
- Probabilistic assertions with Azuma-Hoeffding theorem
- Almost-sure termination by super martingale ranking function

Advantages/Disadvantages

+ Step towards automated analysis of probabilistic programs
+ With real-valued variables and continuous distributions
– Automated discovery only for linear programs



References I

- Chakarov, A., Sankaranarayanan, S.:
  Probabilistic Program Analysis with Martingales.
  In: Computer Aided Verification, Springer (2013) 511–526
- McIver, A., Morgan, C.C.:
  Abstraction, Refinement and Proof for Probabilistic Systems.
  Monographs in Computer Science. Springer (2006)
- Chakarov, A., Sankaranarayanan, S.:
  Expectation Invariants for Probabilistic Program Loops as Fixed Points.
  In: Static Analysis.
  Springer (2014) 85–100

References II

Chung, F., Lu, L.:
Complex Graphs and Networks. Volume 107 of CBMS-NSF.
American Mathematical Society (2006)

Backup Slides for Questions

AST proof not sound:

A probabilistic program which terminates almost surely and has no SMRF:

```
int x := 10; while (x >= 0) { if (flip(0.5)) x++; else x --; }
```

Evaluation example:

- Probabilistic program for Roulette game
- Automatically found Super Martingale: `money`

$\Rightarrow$ Almost sure termination in gamblers ruin