Probabilistic Program Analysis and Concentration of Measure

Part I: *Concentration of Measure*

Sriram Sankaranarayanan

University of Colorado, Boulder

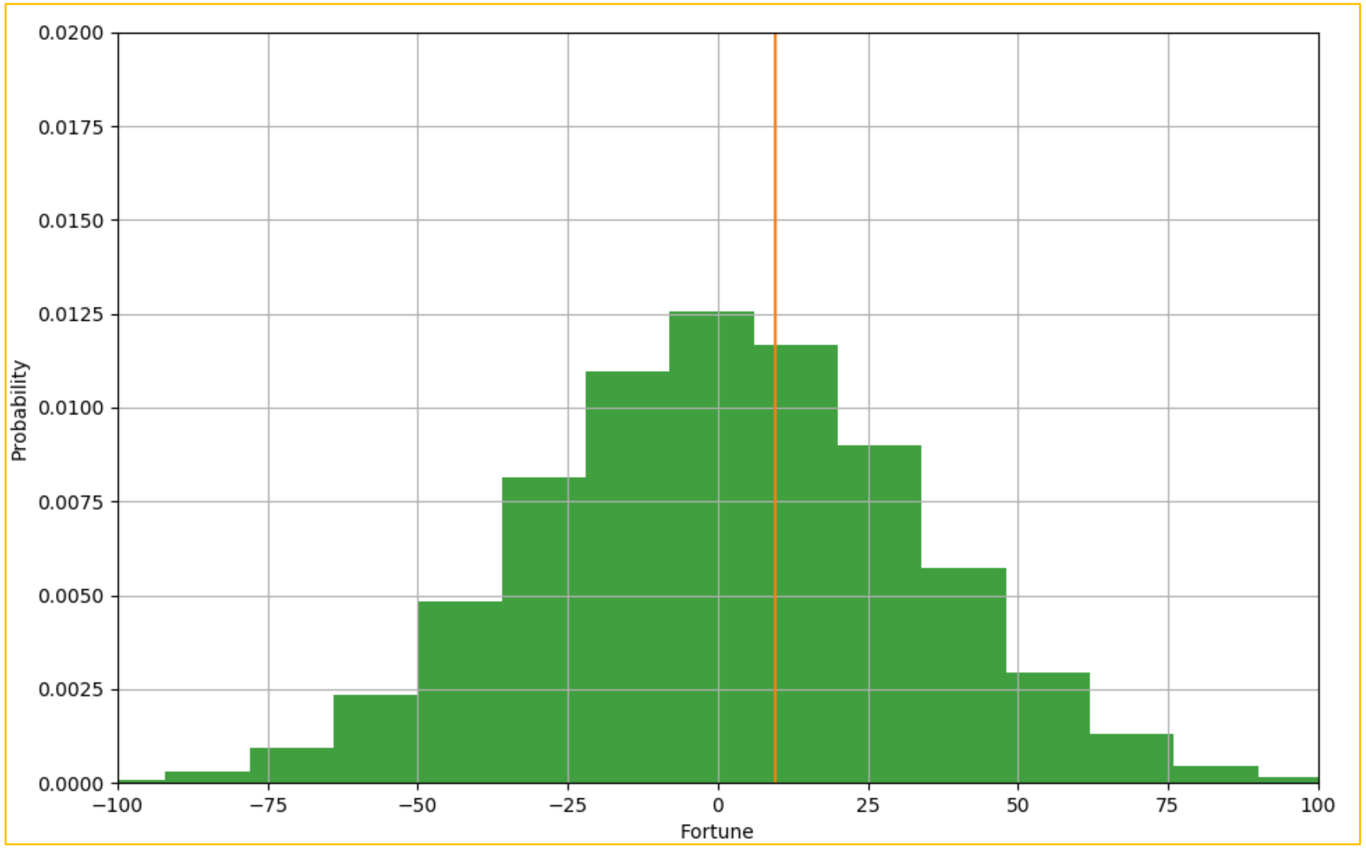
Concentration of Measure: Experiment #1

Heads → Gain one dollar

Repeat 1000 times.

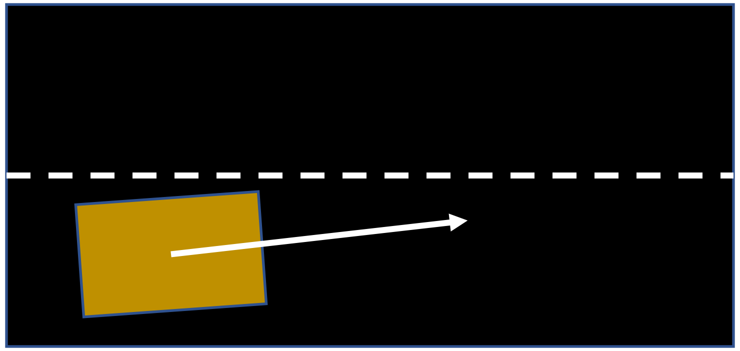
Tails → Lose one dollar

Best Case: + 1000 Dollars
Worst Case: - 1000 Dollars.
Average Case: 0 Dollars.



Concentration of Measure: Experiment #2

Vehicle on a road.



(x, y, θ)

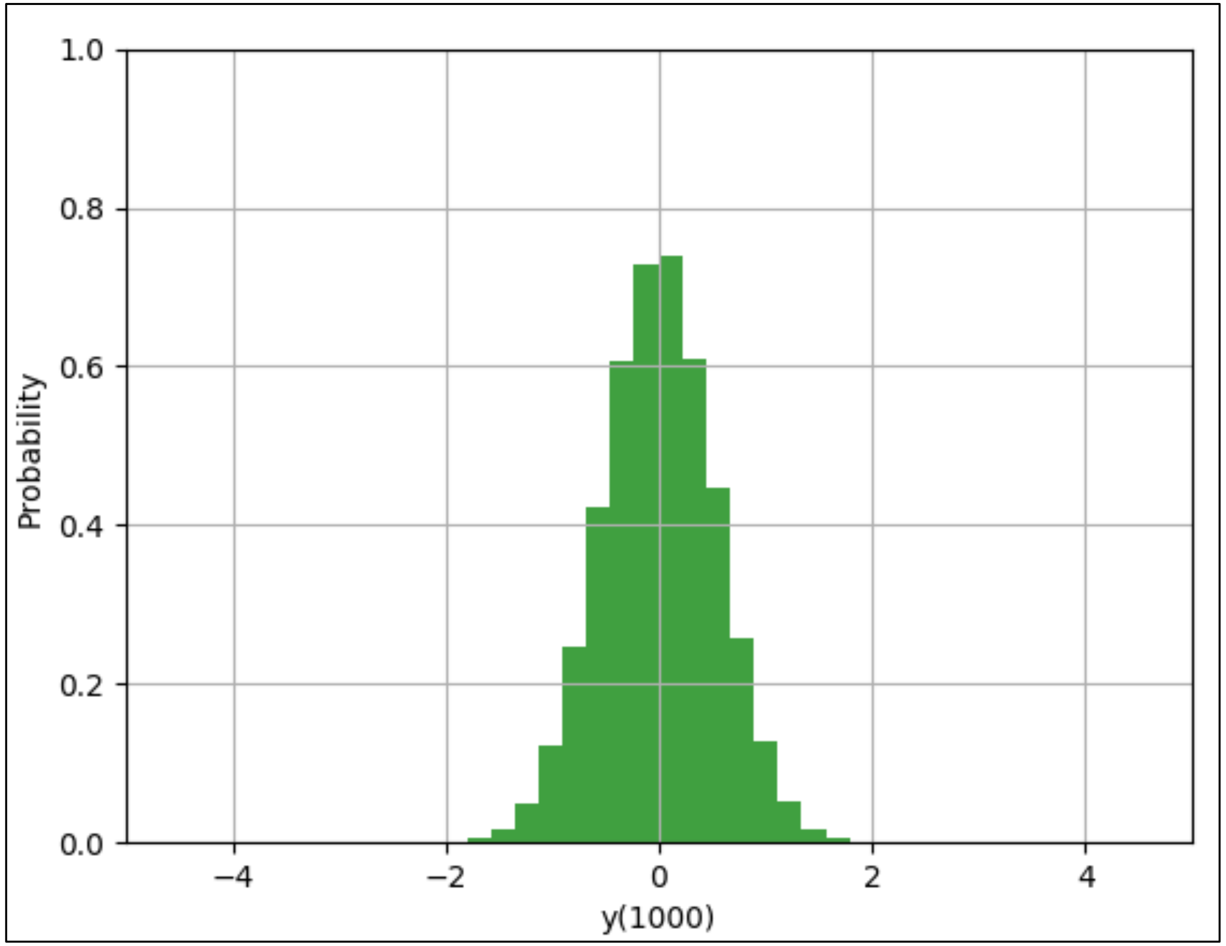
$$\cancel{x(t+1) = x(t) + 0.1\cos(\theta)}$$

$$y(t+1) = y(t) + 0.1\sin(\theta)$$

$$\boxed{\sin(\theta) \simeq \theta}$$

$$\theta(t+1) = 0.8\theta(t) + w$$

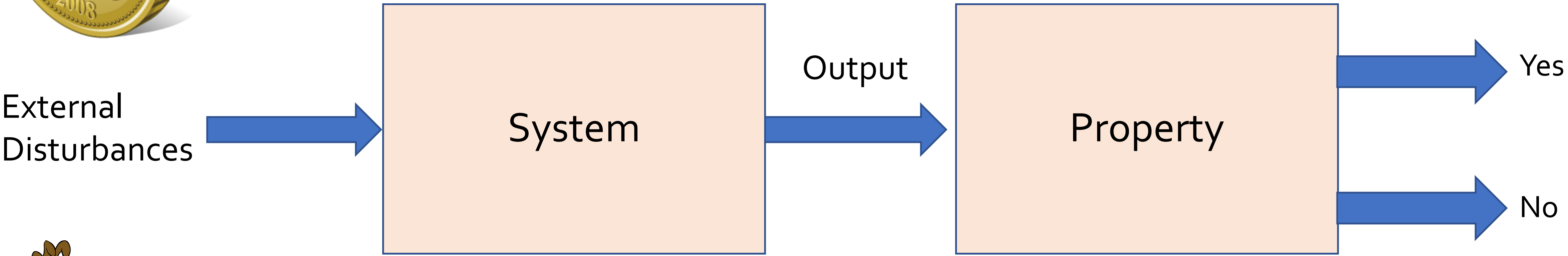
$$w \sim \mathcal{N}(0, 0.1)$$



Systems Acting Under Disturbances

- Stochastic Verification
- Reliability
- Stochastic Controls
- Uncertainty Quantification

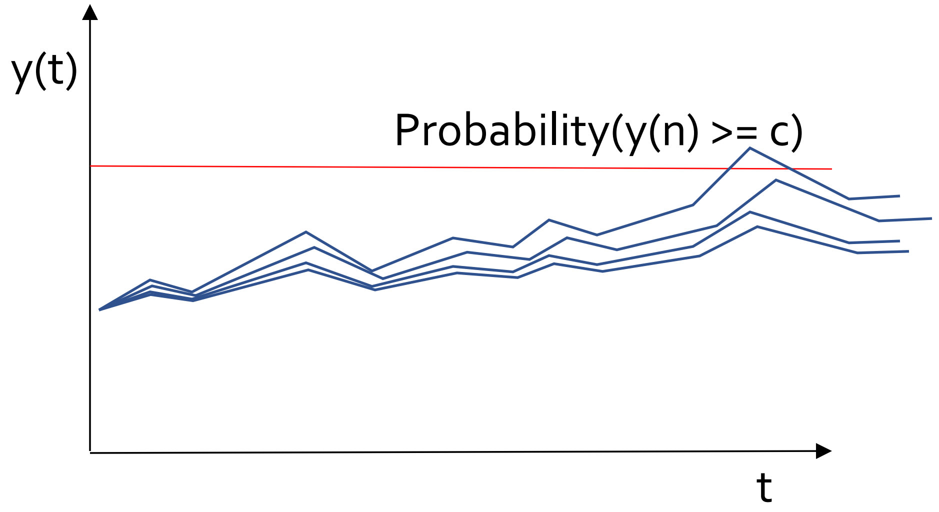
- Artificial Intelligence
  - Uncertainty Representations



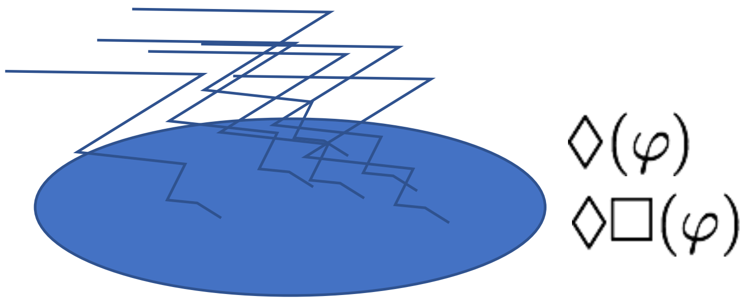
- “Classic” Formal Verification.
- “Set-Valued” Robust Control.

Topics Covered

- Quantitative Reasoning:
  - Prove bounds on probabilities of assertions.
  - Bounds on expectations/moments.



- Qualitative Reasoning:
  - Almost sure termination, recurrence, persistence, ..
  - Limit behavior.



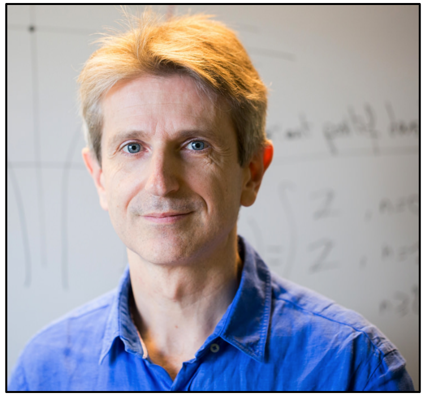
Please ask questions during the talk!

Papers

Aleksandar Chakarov, PhD Thesis, University of Colorado, Boulder, August 2016.

- Aleksandar Chakarov, Yuen-Lam (Vris) Voronin, and Sriram Sankaranarayanan, *Deductive Proofs of Almost Sure Persistence and Recurrence Properties* In TACAS 2016.
- Olivier Bouissou, Eric Goubault, Sylvie Putot, Aleksandar Chakarov, and Sriram Sankaranarayanan, *Uncertainty Propagation using Probabilistic Affine Forms and Concentration of Measure Inequalities.* In TACAS 2016.
- Aleksandar Chakarov, and Sriram Sankaranarayanan, *Probabilistic Program Analysis using Martingales*. In CAV 2013.
- Sriram Sankaranarayanan, Aleksandar Chakarov, and Sumit Gulwani, *Static Analysis for Probabilistic Programs: Inferring Whole Program Properties from Finitely Many Paths* In PLDI 2013.

Co-Authors



Aleksandar Chakarov
Univ. Colorado, Boulder
now at Phase Change

Olivier Bouissou
CEA, now at Mathworks

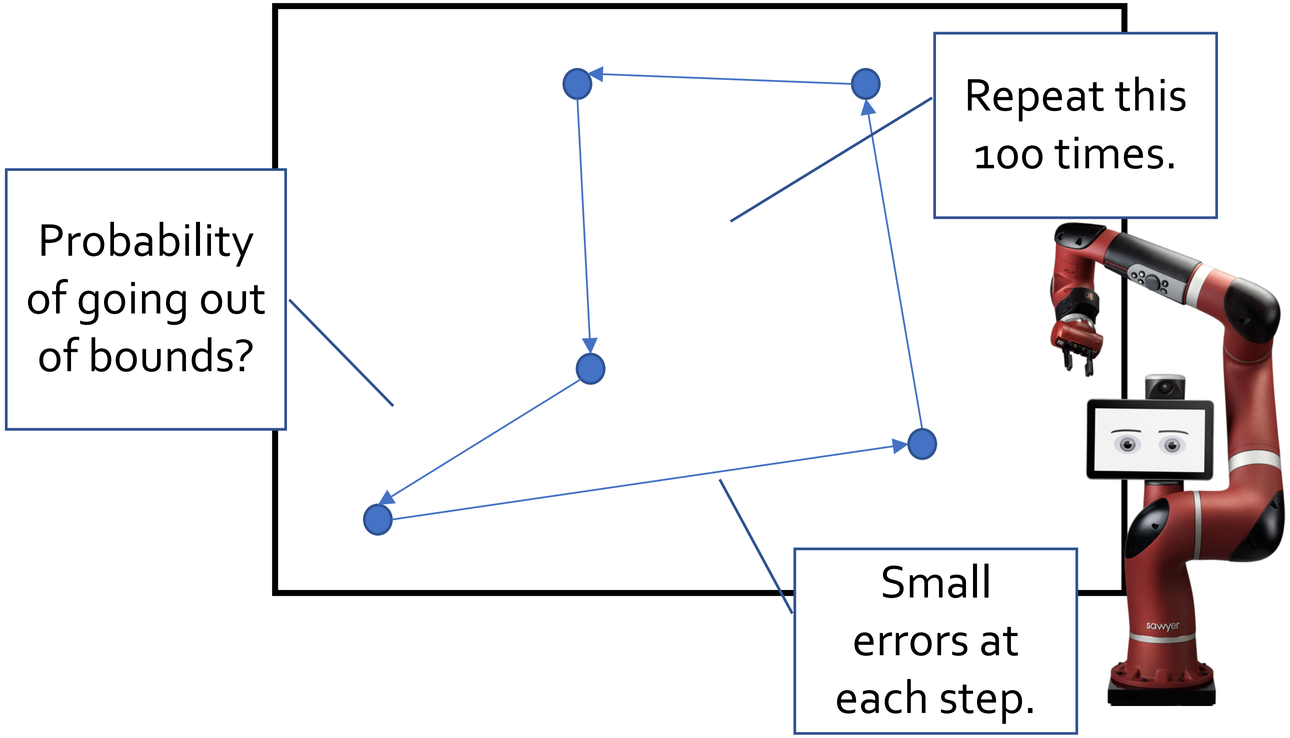
Eric Goubault
Ecole Polytechnique

Sylvie Putot
Ecole Polytechnique

Yuen-Lam Voronin
Univ. Colorado, Boulder

Motivating Examples

Example #1: Repetitive Robot

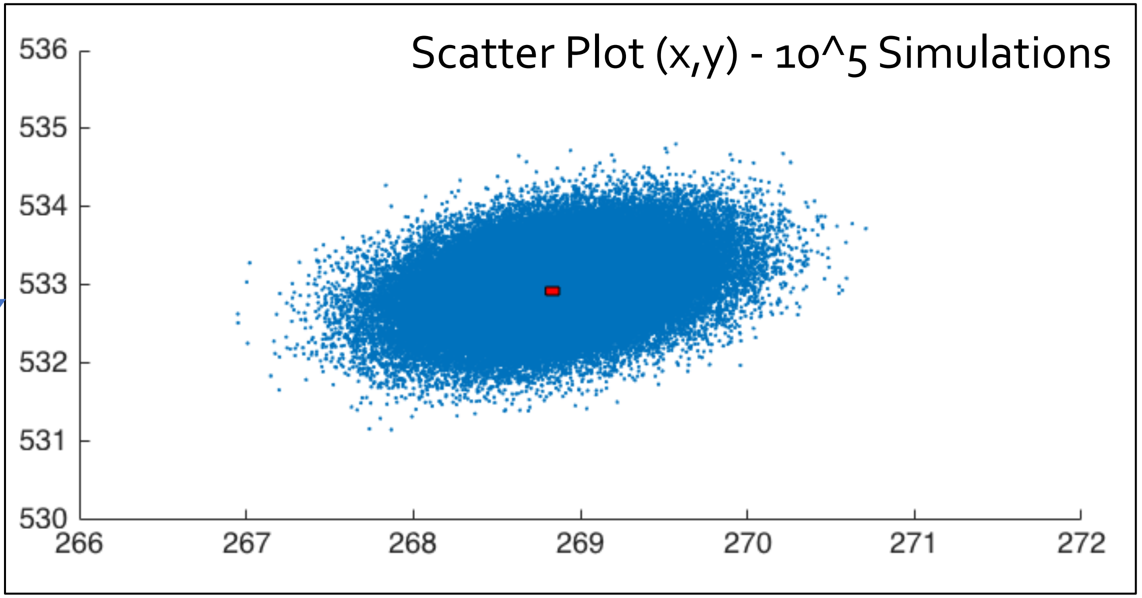


Sawyer Robotic Arm
(rethink robotics)

```
angles = [10, 60, 110, 160, 140, ...
            100, 60, 20, 10, 0]
x := TruncGaussian(0,0.05,-0.5,0.5)
y := TruncGaussian(0, 0.1,-0.5,0.5)
for reps in range(0,100):
    for theta in angles:
        # Distance travelled variation
        d = Uniform(0.98,1.02)
        # Steering angle variation
         t = deg2rad(theta) * (1 + ...
                 TruncGaussian(0,0.01,-0.05,0.05))
        # Move distance d with angle t
        x = x + d * cos(t)
        y = y + d * sin(t)
#Probability that we went too far?
assert(x >= 272)
```

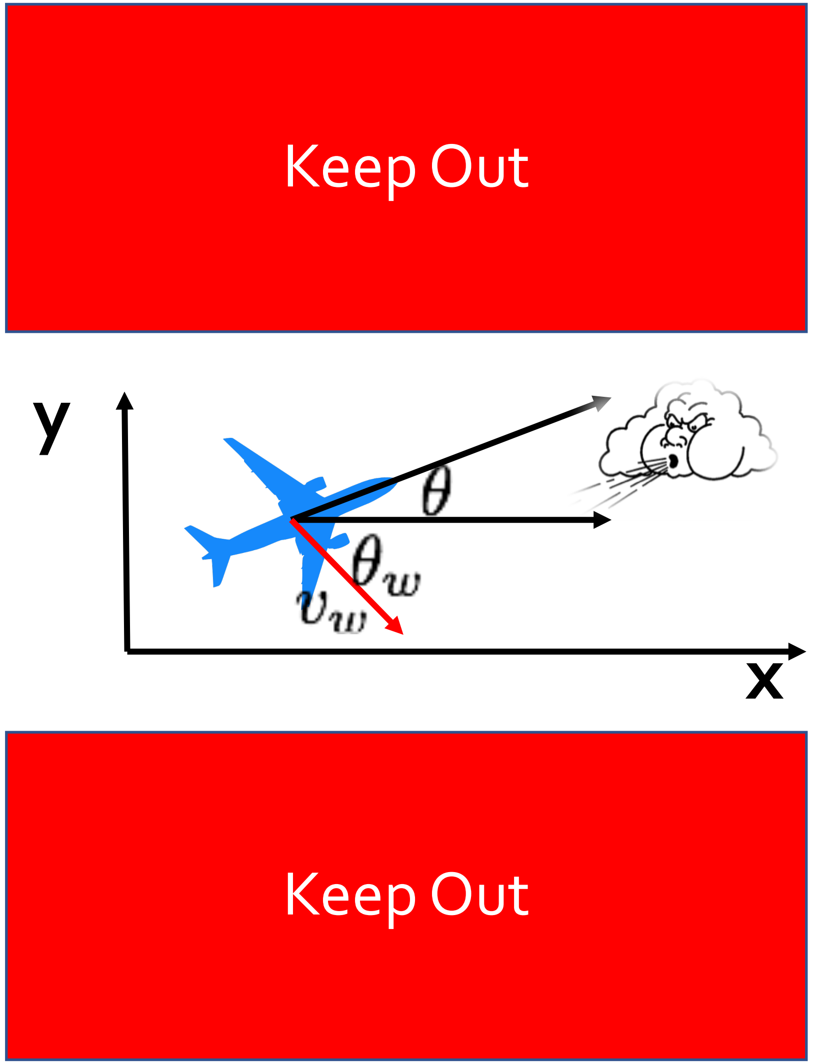
Example #1: Continued

```
angles = [10, 60, 110, 160, 140, ...
              100, 60, 20, 10, 0]
x := TruncGaussian(0,0.05,-0.5,0.5)
y := TruncGaussian(0, 0.1,-0.5,0.5)
for reps in range(0,100):
    for theta in angles:
        # Distance travelled variation
        d = Uniform(0.98,1.02)
        # Steering angle variation
         t = deg2rad(theta) * (1 + ...
                  TruncGaussian(0,0.01,-0.05,0.05))
        # Move distance d with angle t
        x = x + d * cos(t)
        y = y + d * sin(t)
#Probability that we went too far?
assert(x >= 272)
```



$$\mathbb{P}(x \geq 272) \leq ??$$

Example #2: UAV Keep Out Zone



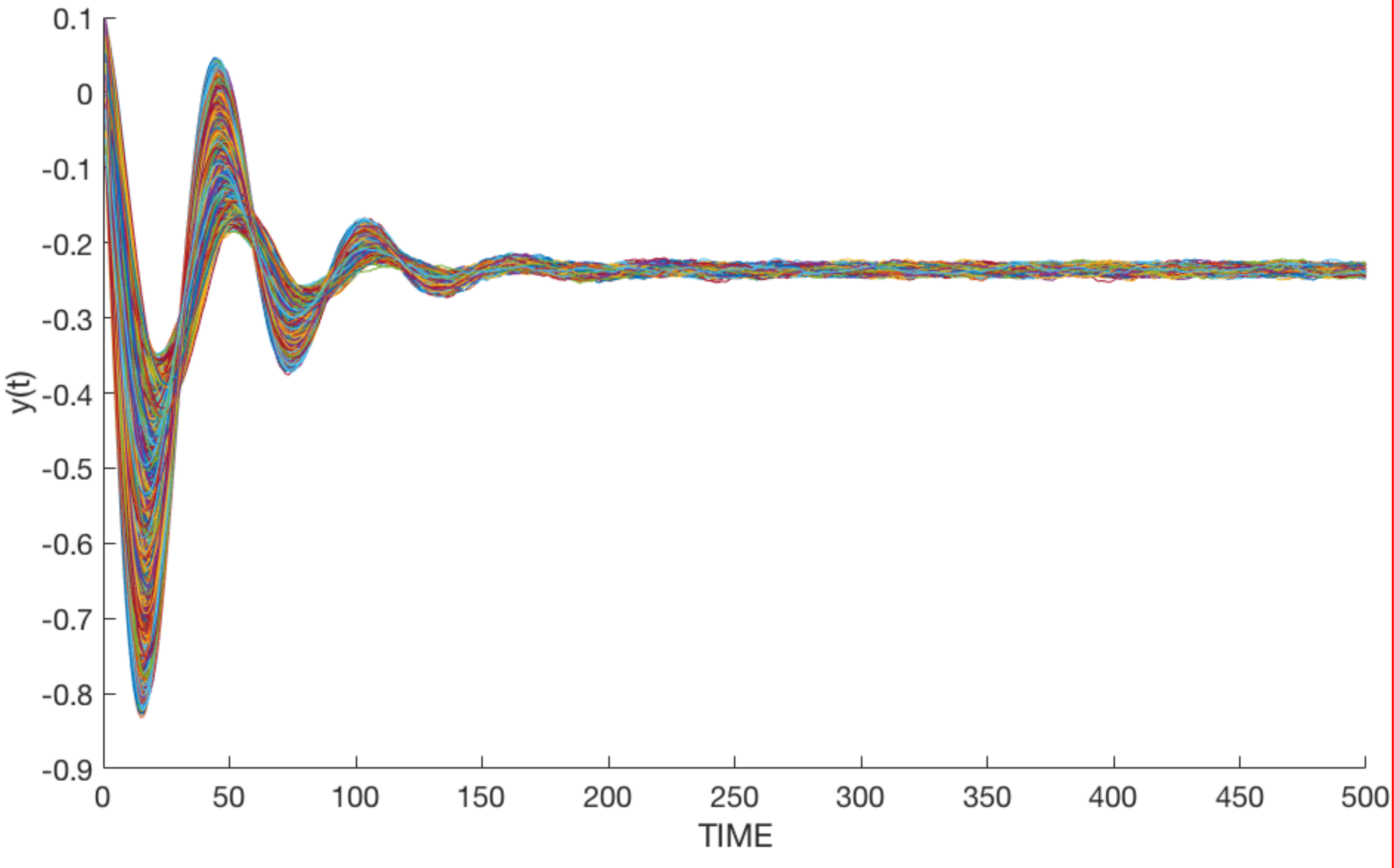
$$y(t+1) = y(t) + 0.1v\sin(\theta(t)) + 0.1v_w\sin(\theta_w(t))$$
$$\theta(t+1) = 0.95\theta(t) - 0.03y(t)$$
$$\theta_w(t+1) \sim 0.6 + w_1$$
$$v_w(t+1) \sim 1 + w_2$$
$$v = 4$$

$$w_1 \in [-0.1, 0.1],\ \mathbb{E}(w_1) = 0,\ \mathbb{E}(w_1^2) = 0.01$$

$$w_2 \in [-0.1, 0.1],\ \mathbb{E}(w_2) = 0,\ \mathbb{E}(w_2^2) = 0.01$$

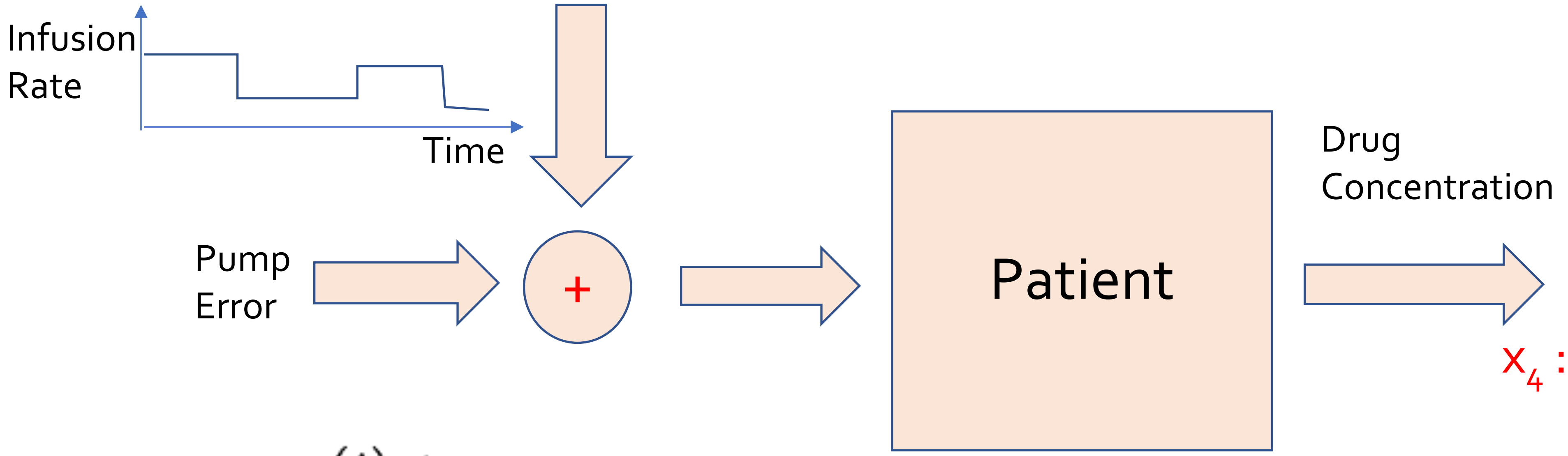
Example #2: UAV Keep Out Zone

```
theta := Uniform(-0.1, 0.1)
y := Uniform(-0.1, 0.1)
for j in range(0, n):
    v := 4
   vw := 1 + random([-0.1, 0.1], 0, 0.01)
   thetaw := 0.6 + random([-0.1, 0.1], 0, 0.01)
    y := y + 0.1  * v * sin(theta) +
              0.1  * vw * sin(thetaw)
   theta := 0.95 * theta – 0.03 * y

Probability( y >= 1.0)
Probability(y <= -1.0)
```



Anesthesia (Fentanyl) Infusion

[McClain+Hug, Fentanyl Kinetics, Clinical Pharmacology & Therapeutics, 28(1):106–114, July 1980.



x_4 : [150, 300] ng/ml

$$u = u(t) + w$$

$$x_1(t+1) = 0.9012 x_1(t) + 0.0304 x_2(t) + 0.0031 x_3(t) + 0.2676 u$$

$$x_2(t+1) = 0.0139 x_1(t) + 0.9857 x_2(t) + 0.002 u$$

$$x_3(t+1) = 0.0015 x_1(t) + 0.9857 x_3(t) + 0.0002 u$$

$$x_4(t) = 0.0838 x_1(t) + 0.0014 x_2(t) + 0.0001 x_3(t) + 0.9117 x_4(t) + 0.012 u$$

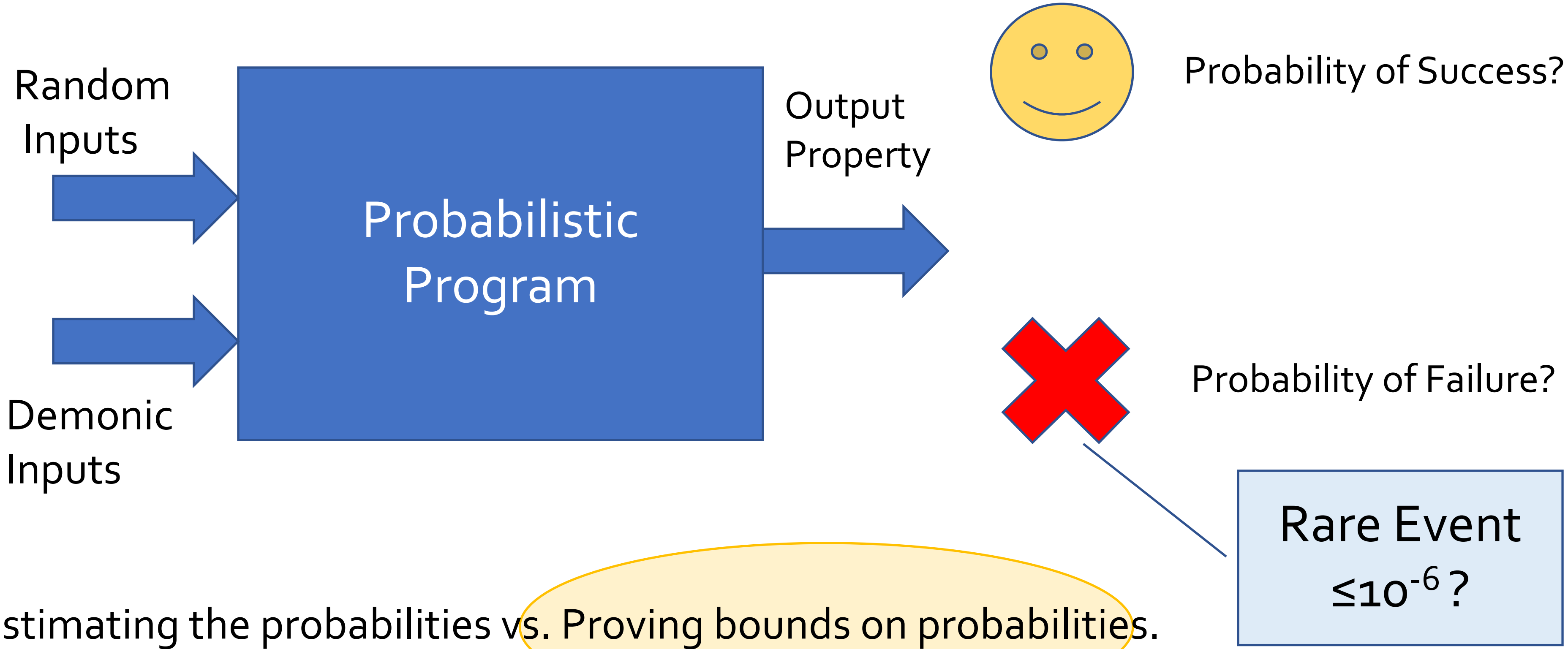
Anesthesia Infusion (Continued)

```
infusionTimings[7] = {20, 15, 15, 15, 15, 15, 45};
double infusionRates[7] = { 3, 3.2, 3.3, 3.4, 3.2, 3.1, 3.0};
Interval e0(-0.4, 0.4), e1(0.0), e2(0.006,0.0064);
for i in range(0, 7):
    currentInfusion= 20.0*infusionRates[i];
    curTime = infusionTimings[i];
    for j in range(0, 40 * infusionTimings[j]):
        e := 1+ randomVariable(e0, e1, e2)
        u := e * currentInfusion
        x1n := 0.9012* x1 + 0.0304 * x2 + 0.0031 * x3
               + 2.676e-1 * u
        x2n := 0.0139* x1 + 0.9857 * x2 + 2e-3*u
        x3n := 0.0015 * x1 + 0.9985 * x3+ 2e-4*u
        x4n := 0.0838 * x1 + 0.0014 * x2 + 0.0001 *x3 +
               0.9117 * x4 + 12e-3 * u
        x1 := x1n;    x2 := x2n;
        x3 := x3;  x4 := x4n
```

$$\mathbb{P}(x_4 \leq 150\mathtt{ng/ml})$$

$$\mathbb{P}(x_4 \geq 300\mathtt{ng/ml})$$

Reasoning about Uncertainty



Estimating the probabilities vs. Proving bounds on probabilities.

Agenda

- Probabilities and Programs.
  - Probabilistic Properties.
- Concentration of Measure Inequalities.
  - Finite executions, “straight line” programs.
- Martingales and more general programs.
  - Pre-expectation calculus.
  - Reasoning about termination.
  - Reasoning about temporal properties.

Programming with Probabilities

- Imperative programs with random number generation.

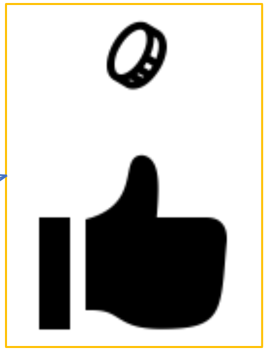
```
real x := Uniform(-1, 1)
real y := Gaussian(2.5, 1.3)
bool b := true
int i := 0
for i in range(0, 100):
   b := Bernoulli(0.5)
   if b:
      x := x + 2 *y + Gaussian(0.5, 1.5)
   else:
      x := 1 – 2.5 * x
      y := 2
assert( x >= y)
```

Random Number Generation Function

Demonic Nondeterminism

- Ignore demonic nondeterminism.
- Focus purely on random variables.

```
real x := Uniform(-1, 1)
real y := Gaussian(2.5, 1.3)
bool b
int i
for i in range(0, 100):
   b := Bernoulli(0.5)
   if b:
      x := x + 2 *y + Gaussian(0.5, 1.5)
else:
      x := 1 - 2.5 * x - Choice(-1, 1)
      y := 2
assert( x >= y)
```



Demon chooses so as to maximize probability of failure

Parametric Nondeterminism

```
real x := Uniform(-1, 1)
real y := Gaussian(2.5, 1.3)
bool b
int i
for i in range(0, 100):
   b := Bernoulli(0.5)
   if b:
      x := x + 2 *y + Gaussian(0.5, 1.5)
else:
      x := 1 - 2.5 * x - RandomVariable([-1,1], [-0.1, 0.1], [0.001, 0.0015])
      y := 2
assert( x >= y)
```

$$w \in [-1, 1]$$

$$\mathbb{E}(w) \in [-0.1, 0, 1]$$

$$\mathbb{E}(w^2) \in [0.001, 0.0015]$$

Probabilistic Program Semantics

```
real x := Uniform(-1, 1)
real y := Gaussian(2.5, 1.3)
bool b := true
int i := 0
for i in range(0, 100):
   b := Bernoulli(0.5)
   if b:
      x := x + 2 *y + Gaussian(0.5, 1.5)
   else:
      x := 1 - 2.5 * x
      y := 2
assert( x >= y)
```

- Probabilistic Program = Markov Process.
- State Variables (x, y, i, b)
- Initial Distribution X_o
- State Update Rule:

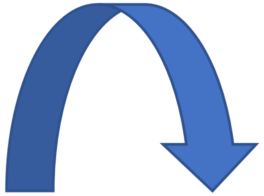
$$(x',y',i',b') = f(x,y,i,b,\underbrace{w_1}_{\text{Gaussian(0.5, 1.5)}},\underbrace{w_2}_{\text{Bernoulli(0.5)}})$$

$$(x',y',i',b') = \begin{cases} (x+2y+w_1,\ y,\ i+1,\ w_2) & i \leq 100 \ \wedge \ w_2 \\ (1-2.5x,\ 2,\ i+1,\ w_2) & i \leq 100 \ \wedge \ \neg w_2 \\ (x,\ y,\ i,\ b) & i > 100 \end{cases}$$

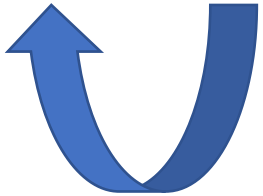
Concentration of Measure

Concentration of Measure: Experiment #1

Heads → Gain one dollar

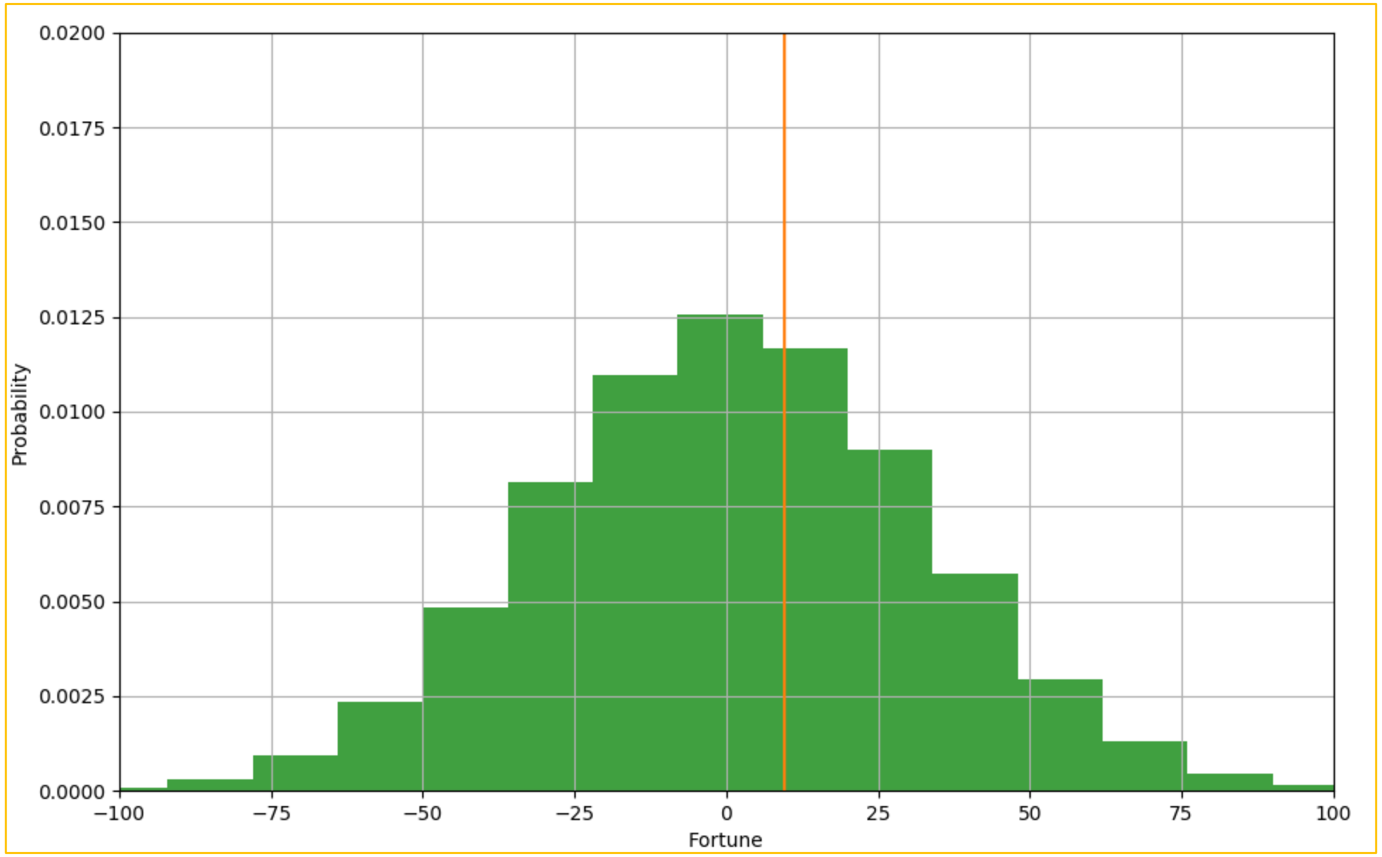


Repeat 1000 times.



Tails → Lose one dollar

Best Case: + 1000 Dollars
Worst Case: - 1000 Dollars.
Average Case: 0 Dollars.

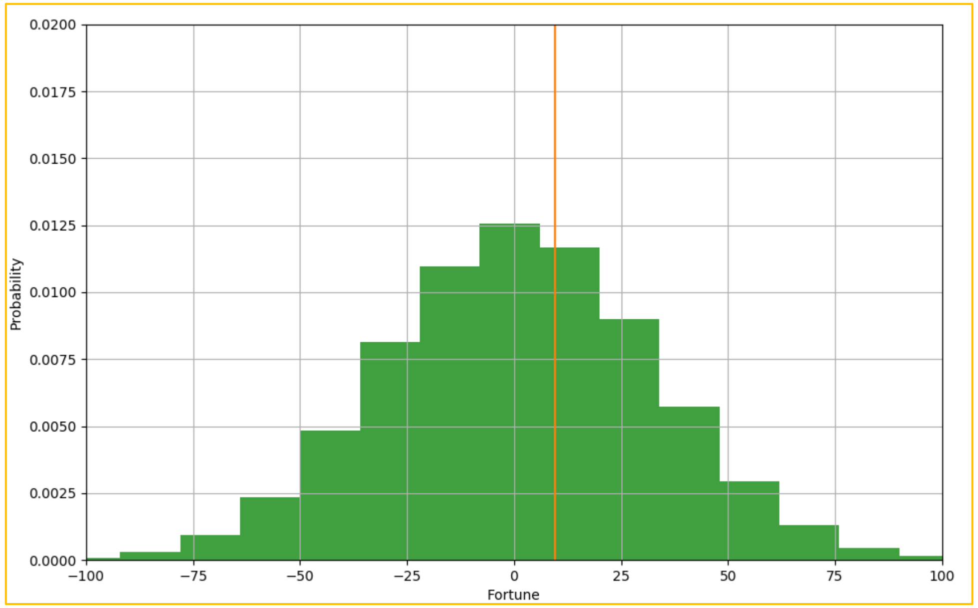


Coin Toss

i.i.d. random variables

$$X_n = C_1 + C_2 + \cdots + C_n$$

$$C_i = \begin{cases} -1 & \text{w.p. } \frac{1}{2} \\ 1 & \text{w.p. } \frac{1}{2} \end{cases}$$

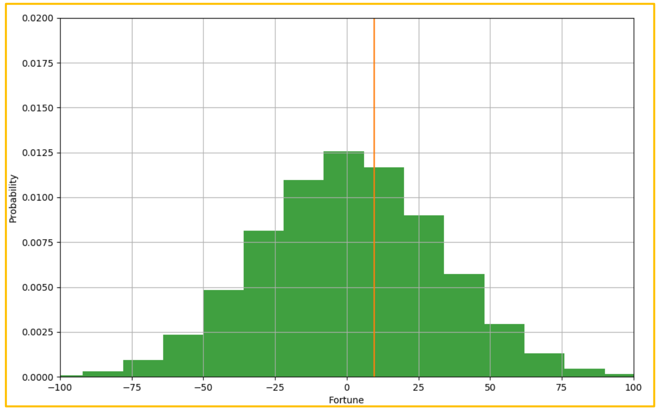


n, j both even or both odd.

What is the distribution of X_n?

$$\mathbb{P}(X_n = j) = \binom{n}{(n+j)/2} \frac{1}{2^n}$$

Coin Toss



$$X_n = C_1 + C_2 + \cdots + C_n$$

$$C_i = \begin{cases} -1 & \text{w.p. } \frac{1}{2} \\ 1 & \text{w.p. } \frac{1}{2} \end{cases}$$

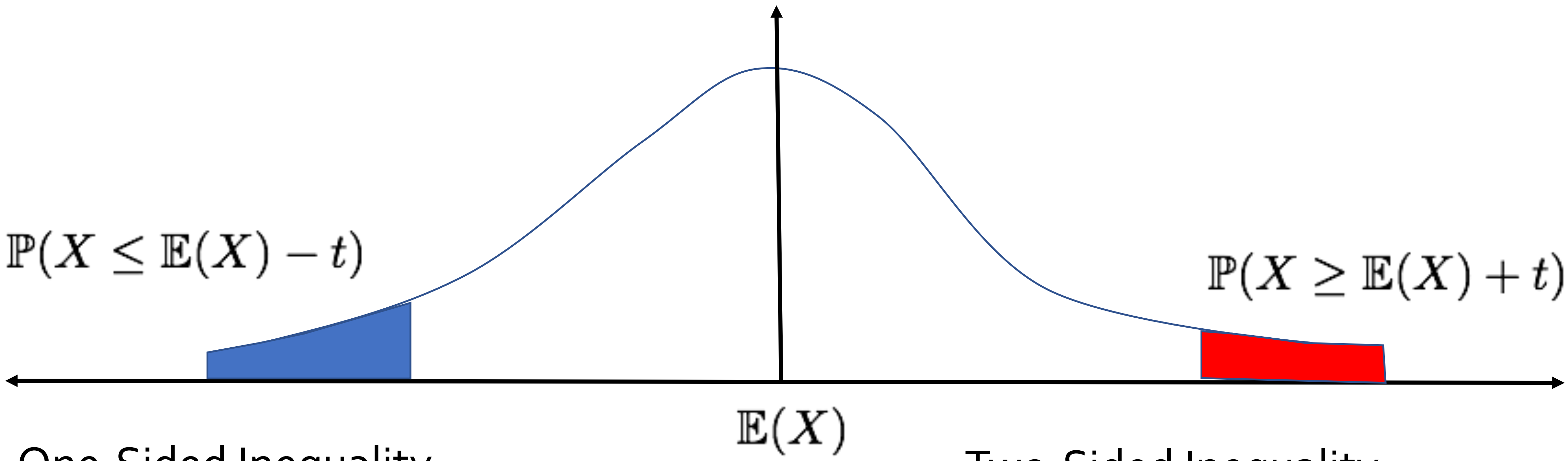
What is the probability X_n >= 100?

$$\sum_{j=100}^{n} \mathbb{P}(X_n = j) = \sum_{j=100}^{n} \binom{n}{(n+j)/2} \frac{1}{2^n}$$

Problem: Not easy to calculate.

Solution: A bound on the probability is good enough.

"Large Deviation" Inequalities



One-Sided Inequality

$$\mathbb{P}(X \geq \mathbb{E}(X) + t) \leq ??$$
$$\mathbb{P}(X \leq \mathbb{E}(X) - t) \leq ??$$

Two-Sided Inequality

$$\mathbb{P}(|X - \mathbb{E}(X)| \geq t) \leq ??$$

Markov Inequality

- Let X be a non-negative random variable.

$$\boxed{\mathbb{P}(X \geq t) \leq \frac{\mathbb{E}(X)}{t}, \quad \text{for all } t > 0}$$

- Corollary (Chebyshev-Cantelli):

$$\boxed{\mathbb{P}(|X - \mathbb{E}(X)| \geq t) \leq \frac{\mathsf{Var}(X)}{t^2}, \quad \text{for all } t > 0}$$

Chernoff-Hoeffding Bounds

$$C_1, \ldots, C_n \in [a, b]$$

1. $C_1, \ldots, C_n \in \{0, 1\}$ independent r.v.
2. $X_n = C_1 + \ldots + C_n$
3. $\mathbb{P}(X_n \geq k)$?

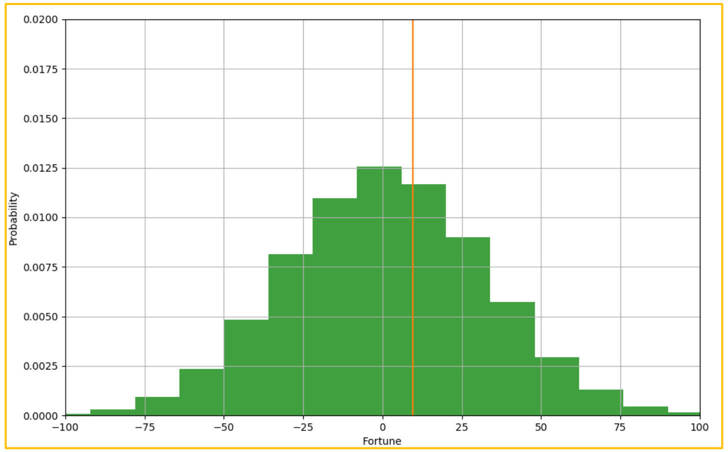
$\mathbb{E}(X_n) = \sum_{j=1}^{n} \mathbb{E}(C_j) =: \mu_n$

$$\boxed{\mathbb{P}(X_n \geq k) \leq \exp\left(\frac{-2(k-\mu_n)^2}{n(b-a)^2}\right)}$$

Theorem (Chernoff' 52, Hoeffding' 63):

$$\mathbb{P}(X_n \geq k) \leq \exp\left(\frac{-2(k-\mu_n)^2}{n}\right)$$

Coin Toss



$$X_n = C_1 + C_2 + \cdots + C_n$$

$$C_i = \begin{cases} -1 & \text{w.p. } \frac{1}{2} \\ 1 & \text{w.p. } \frac{1}{2} \end{cases}$$

What is the probability X_n >= 100?

$$\mu_n = 0,\ k = 100,\ b = 1,\ a = -1$$

$$\mathbb{P}(X_n \geq k) \leq \exp\left(\frac{-2(k-\mu_n)^2}{n(b-a)^2}\right) \qquad \mathbb{P}(X_n \geq 100) \leq \exp\left(\frac{-100^2}{2n}\right)$$

$$\mathbb{P}(X_{1000} \geq 100) \leq \exp(-\frac{10000}{2\times 1000}) \leq 0.006$$

Coin Toss

- Probability bound (0.006) is conservative.
  - Actual value is 10x smaller ~ 5 x10^{-4}
- What information about the coin tosses did we use?

$$C_i = \begin{cases} -1 & \text{w.p. } \frac{1}{2} \\ 1 & \text{w.p. } \frac{1}{2} \end{cases} \quad \Longrightarrow \quad C_i \in [-1, 1],\ \mathbb{E}(C_i) = 0$$

Could we use higher moments to obtain better bounds?

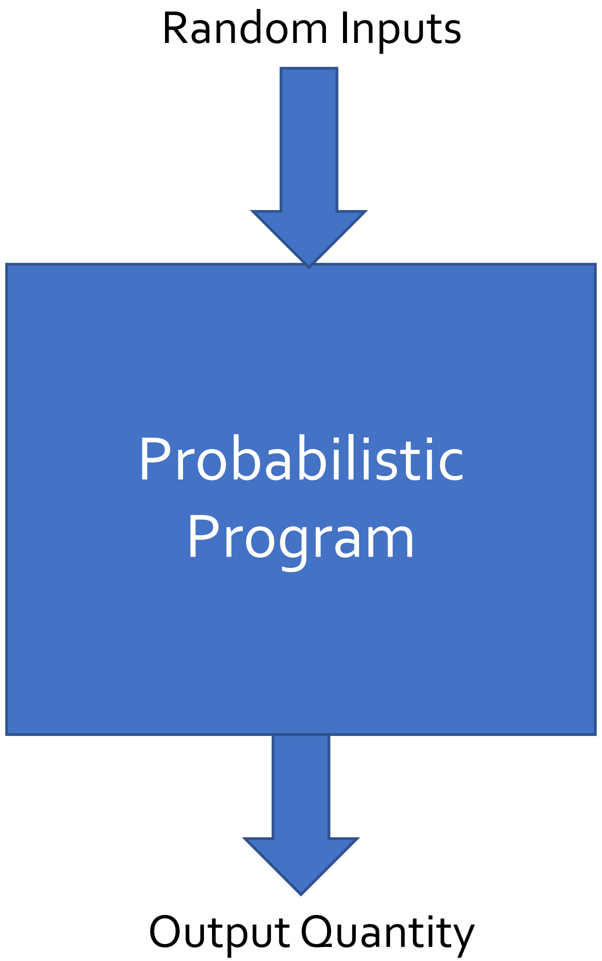
Bernstein Inequality

Extend Chernoff Inequalities with information about moments

$$\mathbb{P}\left(\sum_{i=1}^{n} X_i \geq t\right) \;\leq\; \exp\left(\frac{-t^2}{2\sum \mathbb{E}(X_j^2) + \frac{2}{3}Mt}\right)$$

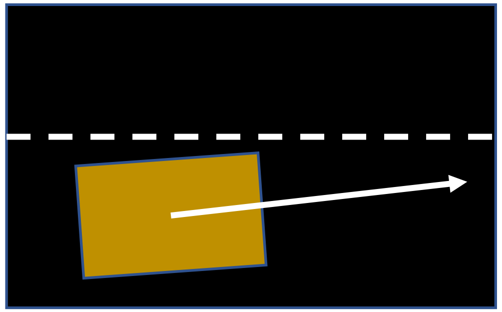
$$M = \max_{i=1}^{n} |X_i|$$

Quantitative Analysis



Lane Keeping Example

(x, y, θ)



$$y(t+1) = y(t) + 0.1\theta$$

$$\theta(t+1) = 0.8\theta(t) + w$$

$$w \in [-0.01, 0.01]$$

$$\mathbb{E}(w) = 0$$

$$\mathbb{E}(w^2) = 0.001$$

```
y := Uniform(-0.01, 0.01)
th := Uniform(-0.01, 0.01)

for i in range(0, n):
    y := y + 0.1 * th
    th := 0.8 * th + randomw()

Probability( y >= 1) <= ??
```

Lane Keeping Example

```
y := Uniform(-0.01, 0.01)
th := Uniform(-0.01, 0.01)

for i in range(0, 10):
    y := y + 0.1 * th
    th := 0.8 * th + randomw()
```

Probability(y >= 0.1) <= ??

$$y[10]: \left(\begin{array}{c} y_0 + 0.357050327 w_0 + \\ 0.4463129088 w_1 + 0.432891136 w_2 + 0.41611392 w_3 + \\ 0.3951424 w_4 + 0.368928 w_5 + 0.33616 w_6 + \\ 0.2952 w_7 + 0.244 w_8 + 0.18 w_9 + 0.1 w_{10} \end{array}\right)$$

"Heterogeneous" Chernoff-Hoeffding Bounds

1. $X = Y_1 + \ldots + Y_n$
2. Y_i are independent r.v.
3. $Y_i \in [a_i, b_i]$
4. $\mu :\ \mathbb{E}(X)$

$$\mathbb{P}(X \geq \mu + t) \ \leq \ \exp\left(\frac{-2t^2}{\sum_{j=1}^{n}(b_i - a_i)^2}\right)$$

$$\mathbb{P}(X \leq \mu - t) \ \leq \ \exp\left(\frac{-2t^2}{\sum_{j=1}^{n}(b_i - a_i)^2}\right)$$

Lane Keeping Example

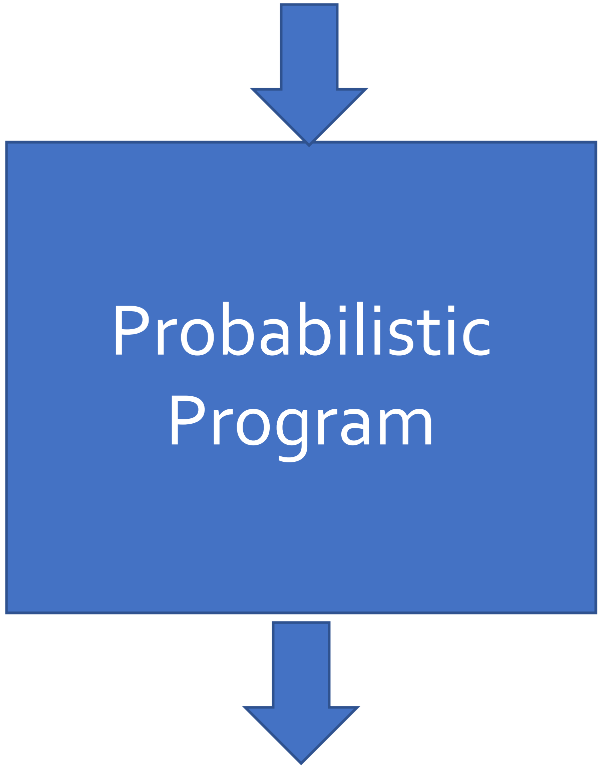
```
y := Uniform(-0.01, 0.01)
th := Uniform(-0.01, 0.01)

for i in range(0, 10):
    y := y + 0.1 * th
    th := 0.8 * th + randomw()

Probability( y >= 0.1) <= ??
```

$$y[10] : \left(\begin{array}{c} y_0 + 0.357050327 w_0 + \\ 0.4463129088 w_1 + 0.432891136 w_2 + 0.41611392 w_3 + \\ 0.3951424 w_4 + 0.368928 w_5 + 0.33616 w_6 + \\ 0.2952 w_7 + 0.244 w_8 + 0.18 w_9 + 0.1 w_{10} \end{array} \right)$$

$$\mathbb{P}(y[10] \geq 0.1) \leq 0.16$$

Problem Setup

Random Inputs (w_0, w_1, … , w_m)



Output Quantity (y)

$$\boxed{y_n = f(w_0, w_1, \ldots, w_m)}$$

$$\boxed{\mathbb{P}(y_n \geq t) \leq ?}$$

Setup

Deterministic Control Flow

```
x := InitialDistribution()

for i in range(0, n):
    w := RandomInputs()
    x := f(i, x, w)

assert(g(x) >= t)
```

$$g(\mathbf{x}_n) = F(\mathbf{x}_0, \mathbf{w}_1, \ldots, \mathbf{w}_n)$$

$$\mathbb{E}(g(\mathbf{x}_n)) = ?$$

$$\mathbb{P}(g(\mathbf{x}) \geq t) \leq ?$$

1. Chernoff-Hoeffding bounds ~ sums of random variables. Not general functions.
2. Need to estimate expectations and possibly higher moments to apply.

Setup

- Chernoff-Hoeffding bounds:
  - Sums of random variables.
  - Extensions to general functions.

```
x := InitialDistribution()

for i in range(0, n):
    w := RandomInputs()
    x := f(i, x, w)

assert(g(x) >= t)
```

- Solution 1: Affine arithmetic and concentration of measure (Bouissou et al. TACAS'16).

- Solution 2: Method of Bounded Differences.

Affine Form Overview

- Affine Form: how program variables depend on the uncertainties.

```
y := Uniform(-0.01, 0.01)
th := Uniform(-0.01, 0.01)

for i in range(0, 10):
      y := y + 0.1 * th
      th := 0.8 * th + randomw()

Probability( y >= 0.1) <= ??
```

$$y[0] = y_0 \qquad \theta[0] = \theta_0$$

$$y[1] = y_0 + 0.1\theta_0$$

$$\theta[1] = 0.8\theta_0 + w_0$$

$$\begin{aligned} y[2] &= y_0 + 0.1\theta_0 + 0.1(0.8\theta_0 + w_0) \\ &= y_0 + 0.18\theta_0 + 0.1w_0 \end{aligned}$$

Affine Form Definition [Figueirido+Stolfi'04, Bouissou et al.]

$$x: \; a_0 + \sum_{i=1}^{n} a_i \boxed{w_i}$$

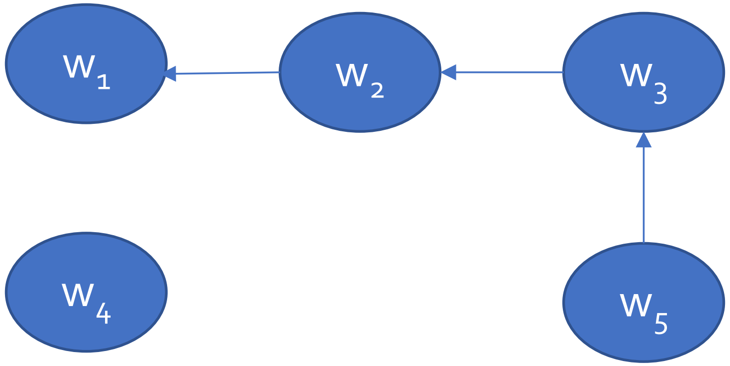
Noise Symbols

$$w_i \in [a_i, b_i]$$

$$\mathbb{E}(w_i) \in [c_i, d_i]$$

$$\mathbb{E}(w_i^2) \in [\ell_i, u_i]$$

$$\mathbb{E}(w_i w_j) \in [f_{ij}, g_{ij}]$$



Functional dependency graph

Computing with Affine Forms

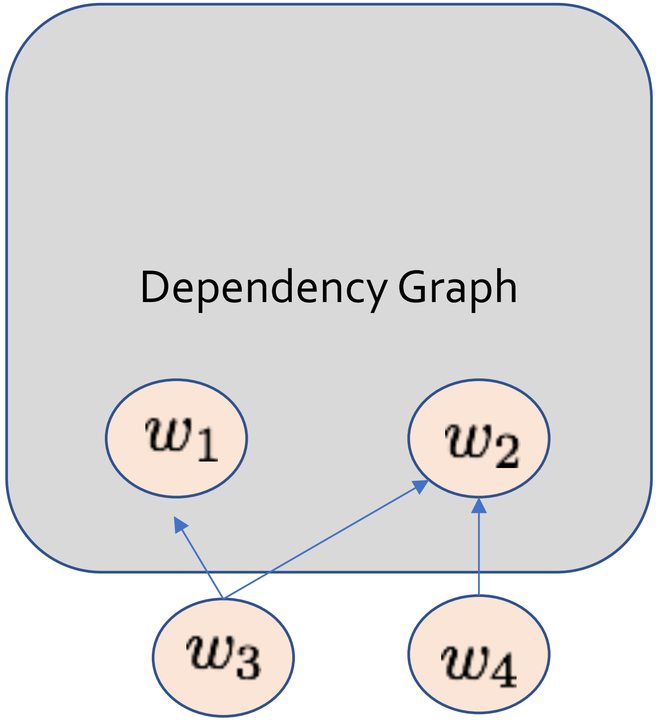
- Linear operations
  - Addition.
  - Multiplication with scalar.
  - Introduction of fresh random variables.
- Nonlinear Operations
  - Multiplication.
  - Division.
  - Sine, cosine, tan, log, exp,…
- Reasoning with affine forms.

Multiplication of Affine Forms

$$x_1 : 2 + 3w_1 + 4w_2$$

$$x_2 : 1 + w_2$$

$$x_1 \times x_2 : 2 + 3w_1 + 6w_2 + 3\underbrace{w_1 w_2}_{w_3} + 4\underbrace{w_2^2}_{w_4}$$



$$\mathbb{E}(w_3) : \mathbb{E}(w_1 w_2)(= \mathbb{E}(w_1)\mathbb{E}(w_2) \text{ if independent})$$

$$\mathbb{E}(w_4) : \mathbb{E}(w_1^2)(\text{second moment of } w_1)$$

Nonlinear Operations

- We will restrict ourselves to smooth operations (continuous + differentiable)

- Let f be a C^k function

$$f(x) : \; x_0 + f'(x_0)(x - x_0) + f''(x_0)\frac{(x - x_0)^2}{2!} + \cdots + \boxed{f^{(k)}(\theta)\frac{(x - x_0)^k}{k!}}$$

x : Affine Form

x_o: E(x)

Fresh noise symbol.

Nonlinear Operation Example

$$\boxed{y = sin(0.2 + w)}$$

$$w \in [0.05, 0.15],\ \mathbb{E}(w) :\ 0.1,\ \mathbb{E}(w^2) = 0.01$$

$$y :\ [0.199986, 0.199987] + [0.9553, 0.9554]w - w_1$$

$$w_1 \in [0, 0.0043],\ \mathbb{E}(w_1) \in [0, 0.00043],\ \mathbb{E}(w_1^2) \in [0, 1.84 \times 10^{-7}]$$

Lane Keeping Example

```
y := Uniform(-0.01, 0.01)
th := Uniform(-0.01, 0.01)

for i in range(0, 10):
    y := y + 0.1 * th
    th := 0.8 * th + randomw()

Probability( y >= 0.1) <= ??
```

$$y[10] : \left(\begin{array}{c} y_0 + 0.357050327 w_0 + \\ 0.4463129088 w_1 + 0.432891136 w_2 + 0.41611392 w_3 + \\ 0.3951424 w_4 + 0.368928 w_5 + 0.33616 w_6 + \\ 0.2952 w_7 + 0.244 w_8 + 0.18 w_9 + 0.1 w_{10} \end{array} \right)$$

$$\mathbb{P}(y[10] \geq 0.1) \leq 0.16$$

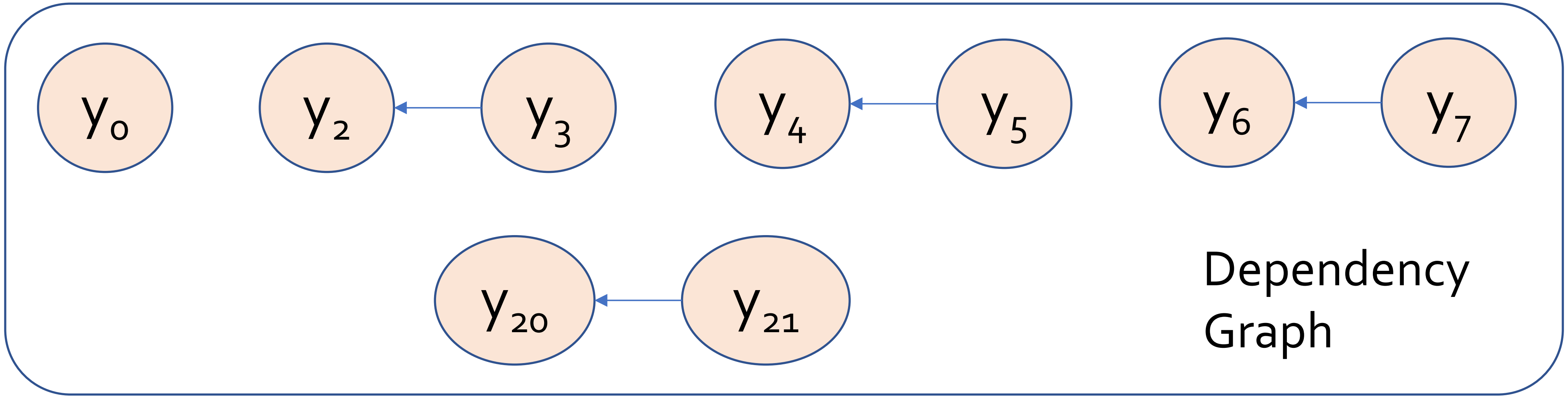
w_1, … , w_{10} are all independent.

Modified Lane Keeping

$$
\begin{aligned}
y = 0 &+ y_0 + 0.1y_2 - 0.1y_3 \\
&+ 0.1y_4 - 0.1y_5 + 0.1y_6 \\
&- 0.1y_7 + 0.1y_8 - 0.1y_9 \\
&+ \cdots \\
&+ 0.1y_{20} - 0.1y_{21}
\end{aligned}
$$

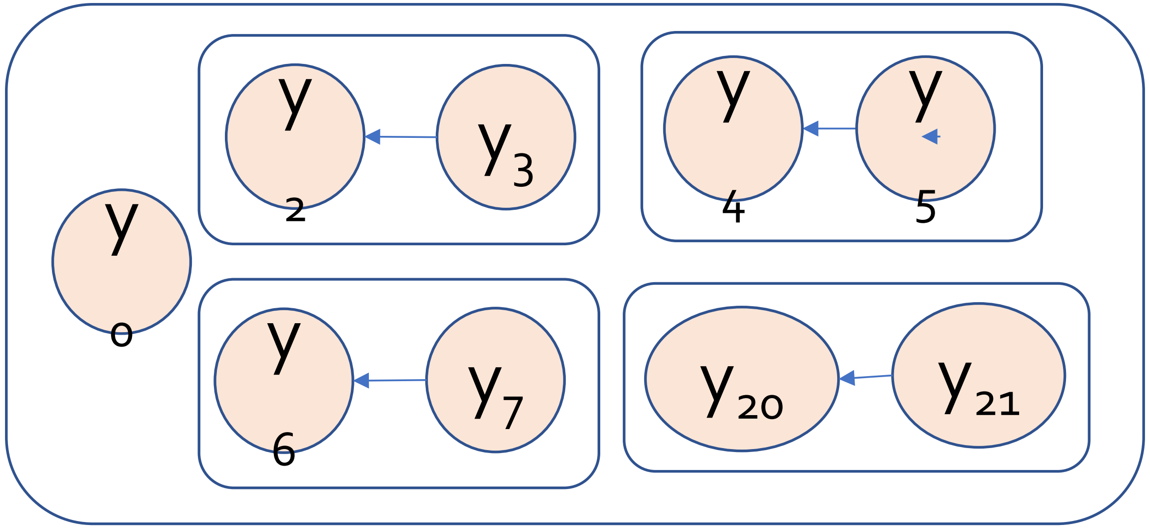
```
y := Uniform(-0.01, 0.01)
th := Uniform(-0.01, 0.01)

for i in range(0, 10):
    y := y + 0.1 * sin(th)
    th := randomw()
```

Probability(y >= 0.1) <= ??



Modified Lane Keeping

$$y = 0 + y_0 + \boxed{0.1y_2 - 0.1y_3} + 0.1y_4 - 0.1y_5 + 0.1y_6$$
$$- 0.1y_7 + 0.1y_8 - 0.1y_9 + \cdots + 0.1y_{20} - 0.1y_{21}$$



Idea:
1. "Compress" connected component to a single noise symbol.
2. Use Chernoff Hoeffding Bounds.

Modified Lane Keeping

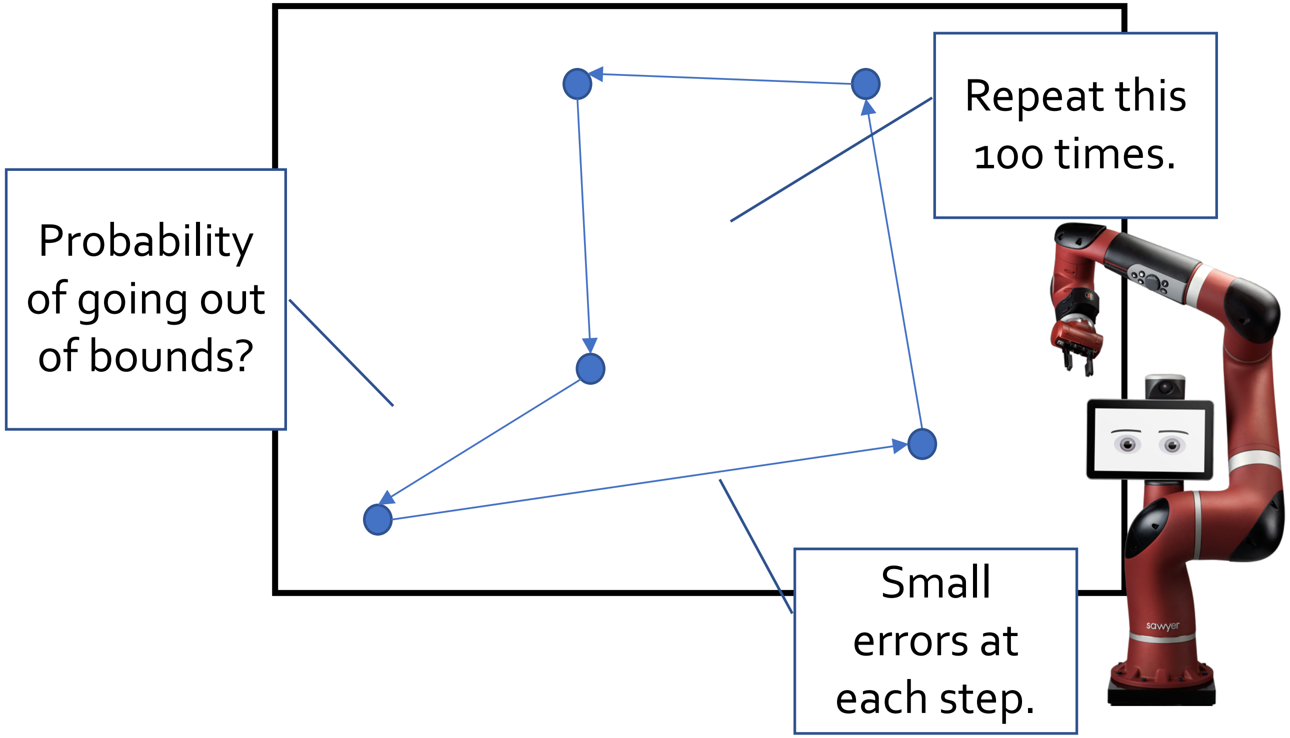
$$\mathbb{P}(y \leq -0.06) \leq 0.006$$
$$\mathbb{P}(y \leq -0.03) \leq 0.16$$
$$\mathbb{P}(y \leq -0.02) \leq 0.45$$
$$\mathbb{P}(y \leq -0.01) \leq 0.82$$
$$\mathbb{P}(y \leq -0.006) \leq 0.96$$

$$\mathbb{P}(y \geq 0.006) \leq 0.96$$
$$\mathbb{P}(y \geq 0.01) \leq 0.82$$
$$\mathbb{P}(y \geq 0.02) \leq 0.45$$
$$\mathbb{P}(y \geq 0.04) \leq 0.082$$
$$\mathbb{P}(y \geq 0.06) \leq 0.006$$

```
y := Uniform(-0.01, 0.01)
th := Uniform(-0.01, 0.01)

for i in range(0, 10):
    y := y + 0.1 * sin(th)
    th := randomw()

Probability( y >= 0.1) <= ??
```

$$\mathbb{P}(y \geq 0.1) = 0$$

Example #1: Repetitive Robot



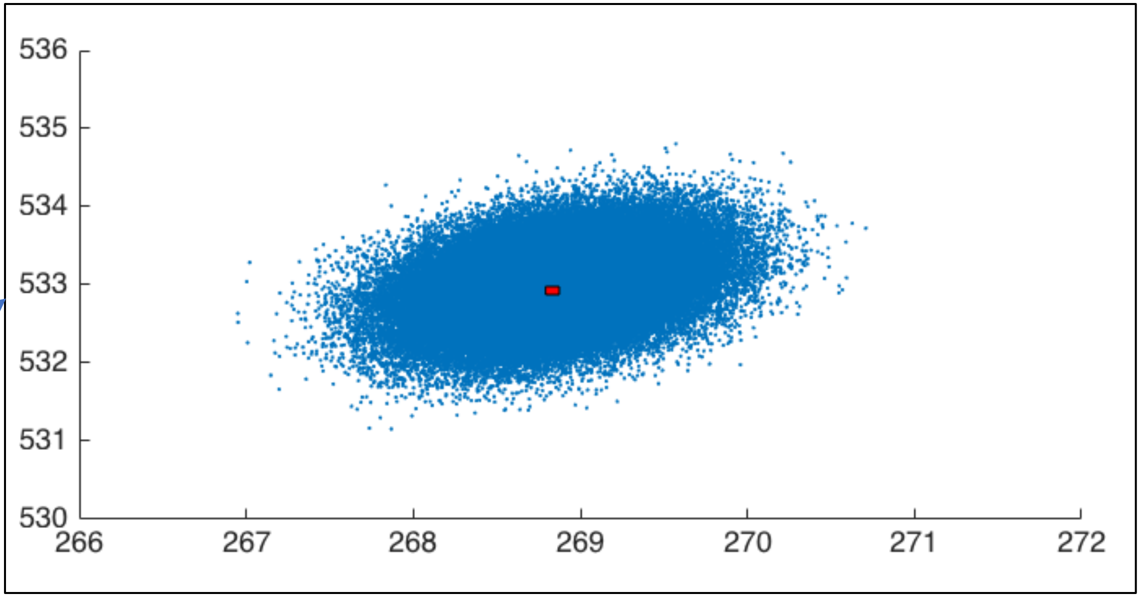
Sawyer Robotic Arm
(rethink robotics)

```
angles = [10, 60, 110, 160, 140, ...
           100, 60, 20, 10, 0]
x := TruncGaussian(0,0.05,-0.5,0.5)
y := TruncGaussian(0, 0.1,-0.5,0.5)
for reps in range(0,100):
    for theta in angles:
        # Distance travelled variation
        d = Uniform(0.98,1.02)
        # Steering angle variation
         t = deg2rad(theta) * (1 + ...
                TruncGaussian(0,0.01,-0.05,0.05))
        # Move distance d with angle t
        x = x + d * cos(t)
        y = y + d * sin(t)
#Probability that we went too far?
assert(x >= 272)
```

Example #1: Continued

```
angles = [10, 60, 110, 160, 140, ...
              100, 60, 20, 10, 0]
x := TruncGaussian(0,0.05,-0.5,0.5)
y := TruncGaussian(0, 0.1,-0.5,0.5)
for reps in range(0,100):
    for theta in angles:
        # Distance travelled variation
        d = Uniform(0.98,1.02)
        # Steering angle variation
         t = deg2rad(theta) * (1 + ...
                  TruncGaussian(0,0.01,-0.05,0.05))
        # Move distance d with angle t
        x = x + d * cos(t)
        y = y + d * sin(t)
#Probability that we went too far?
assert(x >= 272)
```

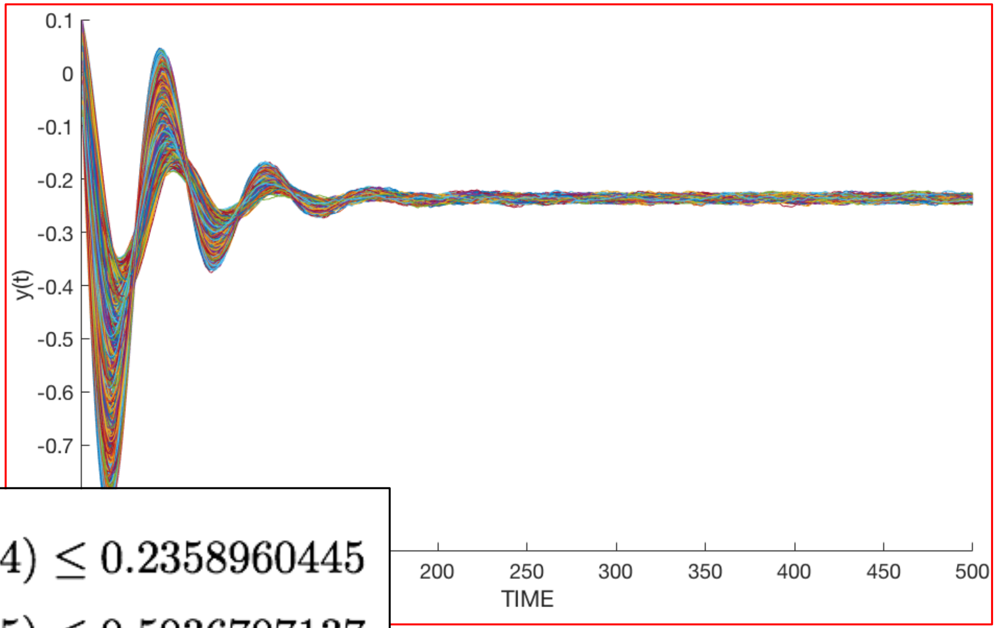
Scatter Plot (x,y) - 10^5 Simulations



$$\mathbb{P}(x \geq 272) \leq 6.2 \times 10^{-7}$$

Example #2: UAV Keep Out Zone

```
theta := Uniform(-0.1, 0.1)
y := Uniform(-0.1, 0.1)
for j in range(0, n):
    v := 4
   vw := 1 + random([-0.1, 0.1], 0, 0.01)
   thetaw := 0.6 + random([-0.1, 0.1], 0, 0.01)
    y := y + 0.1  *  v  *  sin(theta) +
               0.1  *  vw  *  sin(thetaw)
   theta := 0.95  *  theta – 0.03  *  y

Probability( y >= 1.0)
Probability(y <= -1.0)
```



$$\mathbb{P}(y \leq -0.6138812074) \leq 0.2358960445$$
$$\mathbb{P}(y \leq -0.4102788125) \leq 0.5036707137$$
$$\mathbb{P}(y \leq -0.2881173756) \leq 0.7921352709$$

$$\mathbb{P}(y \geq -0.1659559387) \leq 0.9176758046$$
$$\mathbb{P}(y \geq -0.0030740228) \leq 0.5036707137$$
$$\mathbb{P}(y \geq 0.1190874141) \leq 0.3195611154$$
$$\mathbb{P}(y \geq 0.2005283721) \leq 0.2358960445$$

Anesthesia Infusion

```
infusionTimings[7] = {20, 15, 15, 15, 15, 15, 45};
double infusionRates[7] = { 3, 3.2, 3.3, 3.4, 3.2, 3.1, 3.0};
Interval e0(-0.4, 0.4), e1(0.0), e2(0.006,0.0064);
for i in range(0, 7):
    currentInfusion= 20.0*infusionRates[i];
    curTime = infusionTimings[i];
    for j in range(0, 40 * infusionTimings[j]):
         e  : = 1+ randomVariable(e0, e1, e2)
         u  : = e * currentInfusion
          x1n : = 0.9012* x1 + 0.0304 * x2 + 0.0031 * x3
                + 2.676e-1 * u
          x2n := 0.0139* x1 + 0.9857 * x2 + 2e-3*u
          x3n := 0.0015 * x1 + 0.9985 * x3+ 2e-4*u
          x4n := 0.0838 * x1 + 0.0014 * x2 + 0.0001 *x3 +
                0.9117 * x4 + 12e-3 * u
          x1 := x1n;     x2 := x2n;
          x3 := x3;  x4 := x4n
```

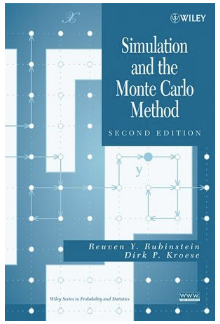
$$\mathbb{P}(x_4 \geq 300\mathtt{ng/ml}) \leq 7 \times 10^{-13}$$

$$\mathbb{P}(x_4 \geq 150\mathtt{ng/ml}) \leq 10^{-23}$$

Concluding Thoughts

Related Approaches

- Monte Carlo Methods
  - Statistical model checking. [Younes+Simmons, Jha et al, Clarke et al.]
  - Importance Sampling [Legay et al.]
  - Semantic Importance Sampling [Hansen et al. TACAS 2015, RV2016]



- Volume Computation
  - Solve the integration exactly (expensive) [Geldenhuys et al, S et al.]
  - Abstract the program by discretizing state space [Abate et al., PRISM]
  - Abstract the distribution by discretizing [Monniaux, Bouissou et al.]

  [Cousot + Monerau]

  - Polynomial-Time Approximation [Chistikov et al. TACAS 2015]

Challenge #1: Representing Nonlinear Computations

How do you represent nonlinear computations?

```
theta := Uniform(-0.1, 0.1)
y := Uniform(-0.1, 0.1)
for j in range(0, n):
    v := 4
   vw := 1 + random([-0.1, 0.1], 0, 0.01)
   thetaw := 0.6 + random([-0.1, 0.1], 0, 0.01)
    y := y + 0.1 * v * sin(theta) +
              0.1 * vw * sin(thetaw)
   theta := 0.95 * theta – 0.03 * y

Probability( y >= 1.0)
Probability(y <= -1.0)
```

Option 1: Affine Forms.

- Approximations create dependencies.

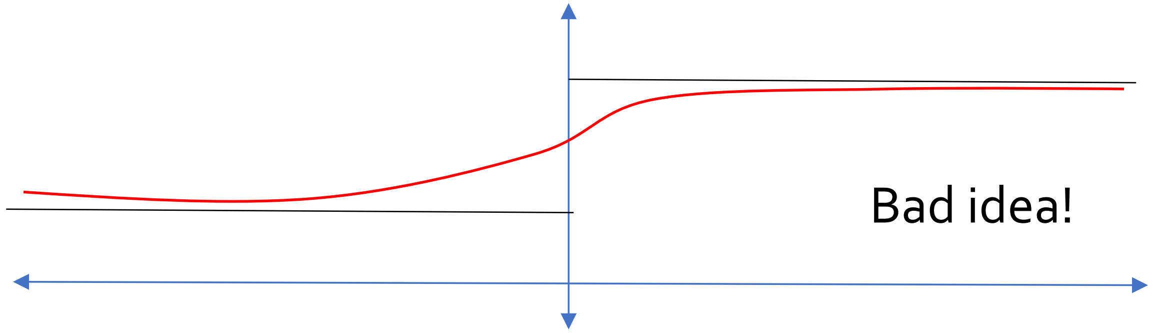
Option 2: Nonlinear Forms.

- Keeps random variables independent.
- Hard to reason with.

Challenge #2: Conditional Branches

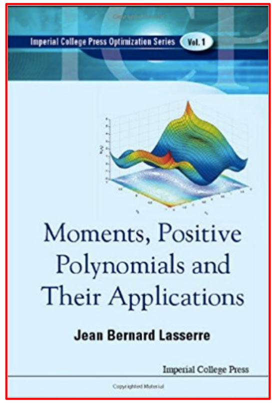
```
theta := Uniform(-0.1, 0.1)
y := Uniform(-0.1, 0.1)
for j in range(0, n):
    v := 4
   vw := 1 + random([-0.1, 0.1], 0, 0.01)
   thetaw := 0.6 + random([-0.1, 0.1], 0, 0.01)
    y := y + 0.1 * v * sin(theta) +
             0.1 * vw * sin(thetaw)
   if y >= 0.1
           theta := theta – 0.1
   if y <= - 0.1
           theta := theta + 0.1
Probability( y >= 1.0)
Probability(y <= -1.0)
```

Approach # 1: Smoothing the Indicator Function.



Approach #2: Moment method.

- Bounds using the problem of moments.
- "Design your own" inequalities.



Probabilistic Program Analysis and Concentration of Measure

Part II: *Martingale*

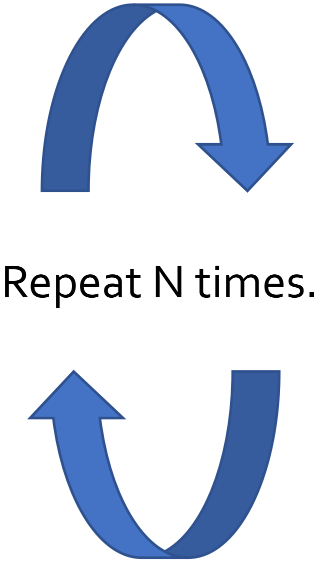
Sriram Sankaranarayanan

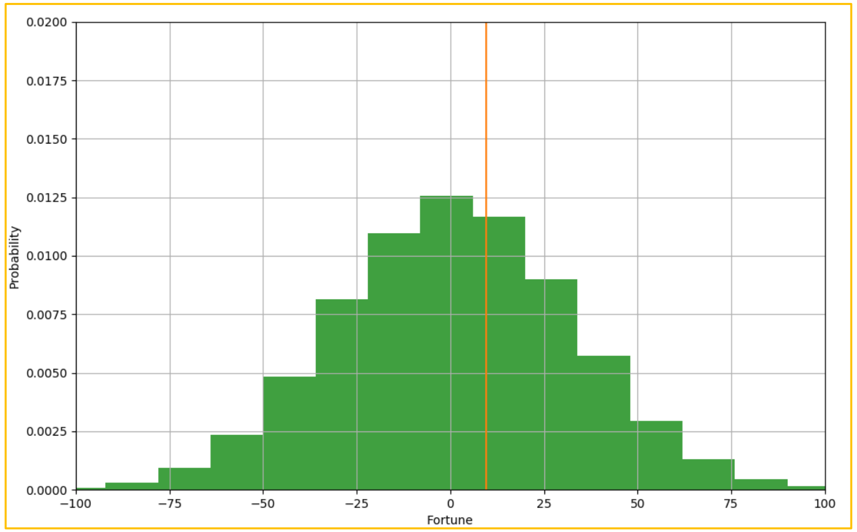
University of Colorado, Boulder

Concentration of Measure: Experiment #1

Heads → Gain one dollar

Tails → Lose one dollar





At some point in the experiment:

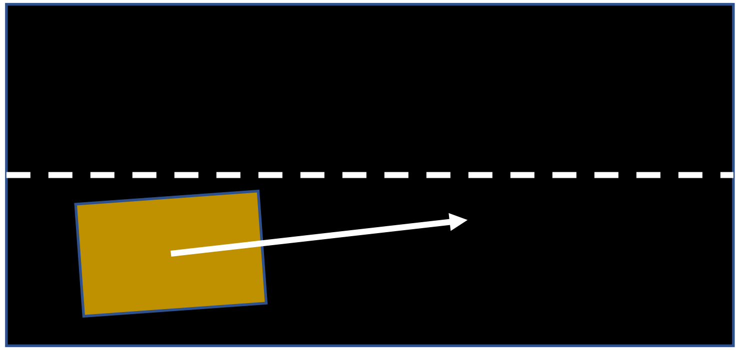
- I have won X_i dollars thus far.
- If I toss once more, how much do I expect to have?

$$\begin{aligned}\mathbb{E}(X_{i+1} \mid X_i) &= \frac{1}{2}(X_i + 1) + \frac{1}{2}(X_i - 1) \\ &= X_i\end{aligned}$$

Expected fortune in next step = fortune in current step.

Concentration of Measure: Experiment #2

Vehicle on a road. (x, y, θ)



$$y(t+1) = y(t) + 0.1\theta$$
$$\theta(t+1) = 0.99\theta(t) + w$$
$$w \in [-0.01, 0.01]$$
$$\mathbb{E}(w) = 0$$

$$\boxed{M(t) : \; y(t) + 10\theta(t)}$$

$$\begin{aligned}\mathbb{E}(M(t+1) \mid y(t), \theta(t)) &= \mathbb{E}\left(\boxed{y(t) + 0.1\theta(t)} + 10(\boxed{0.99\theta(t) + w})\right) \\ &= y(t) + 0.1\theta(t) + 9.9\theta(t) + \mathbb{E}(w) \\ &= y(t) + 10\theta(t) \;\; = M(t)\end{aligned}$$

Expected value in next step = value in current step.

Conditional Expectation

$$\mathbb{E}(X \mid Y) := f(y)$$

$$\mathbb{E}(X \mid Y) = \lambda y.\ \mathbb{E}(X|Y = y)$$

P(Y=y) > 0

$$\sum_{x \in \mathcal{X}} x\mathbb{P}(X = x|Y = y) = \sum_{x \in \mathcal{X}} x \frac{P(X = x, Y = y)}{\mathbb{P}(Y = y)}$$

$$\frac{1}{f_Y(y)} \int_x x f_{X,Y}(x, y) dx$$

Martingale

Martingale is a special kind of stochastic process.

$$X_0,\ X_1,\ X_2,\ \ldots$$

$$\mathbb{E}(X_{i+1} \mid X_i, \ldots, X_0) = X_i$$

$$\boxed{f(x_i, \ldots, x_0)} \qquad \boxed{x_i}$$

Revisit Experiment #1 and #2 slides now!

Super/SubMartingales

Supermartingale: $\mathbb{E}(X_{i+1} \mid X_i, \ldots, X_0) \leq X_i$

Submartingale: $\mathbb{E}(X_{i+1} \mid X_i, \ldots, X_0) \geq X_i$

First Properties of (Super) Martingales

$X_0, \ldots, X_n$ Martingale

$$\mathbb{E}(X_n) = \mathbb{E}(X_0)$$

$X_0, \ldots, X_n$ Supermartingale

$$\mathbb{E}(X_n) \leq \mathbb{E}(X_0)$$

“Adapted” Martingales

$$X_1, \ldots, X_n$$

$$\mathbb{E}(f(X_{i+1}) \mid X_i, \ldots, X_1) = f(X_i)$$

Why Martingales?

- Quantitative: Concentration of measure involving martingales.
- Qualitative: Convergence theorems and proofs of temporal properties.

Martingales and Concentration of Measure (Azuma’s Inequality).

Lipschitz Condition

$$X_0,\ X_1,\ X_2,\ \cdots$$

Lipschitz (Bounded Difference) Condition:

$$|X_i - X_{i-1}| \le c_i,\ \ i > 0$$

Azuma's Inequality for Martingales

$X_0, \ldots, X_n$ stochastic process.

Lipschitz Condition: $|X_i - X_i| \leq c_i$

Supermartingale: $$\mathbb{P}(X_n \geq \mathbb{E}(X_n) + t) \leq \exp\left(\frac{-t^2}{2\sum_{i=1}^{n} c_i^2}\right)$$

Submartingale: $$\mathbb{P}(X_n \leq \mathbb{E}(X_n) - t) \leq \exp\left(\frac{-t^2}{2\sum_{i=1}^{n} c_i^2}\right)$$

Coin Toss Experiment

$$X_n = \sum_{j=0}^{n} C_j, \qquad C_j : \begin{cases} -1 & \text{with prob. } \frac{1}{2} \\ 1 & \text{with prob. } \frac{1}{2} \end{cases}$$

Lipschitz Condition: $\boxed{|X_n - X_{n-1}| \leq 2}$

Chernoff-Hoeffding:

$$\boxed{\mathbb{P}(X_n \geq t) \ \leq \ \exp\left(\frac{-t^2}{2n}\right)}$$

Azuma theorem:

$$\mathbb{P}(X_n \geq t) \leq \exp\left(\frac{-t^2}{8n}\right)$$

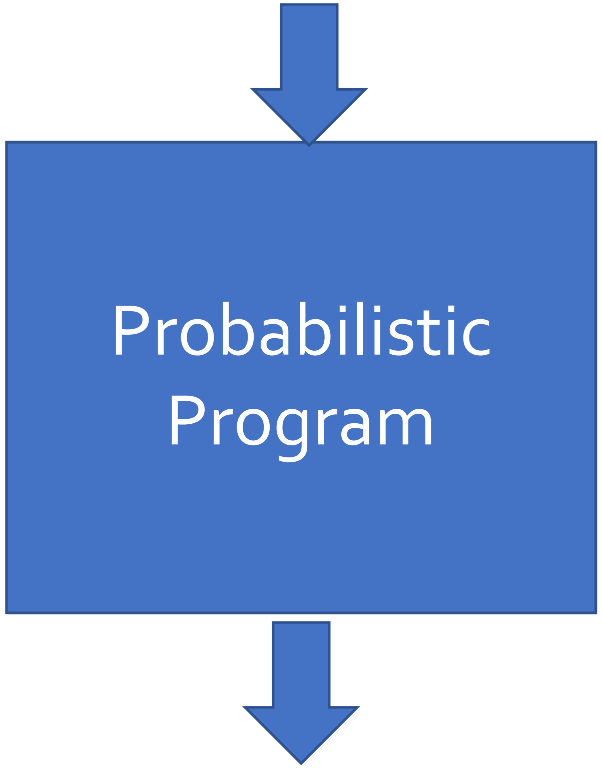
$$\mathbb{P}(X_n \leq t) \leq \exp\left(\frac{-t^2}{8n}\right)$$

Azuma theorem:
No independence assumption.

Doob Martingale or the Method of Bounded Differences

Problem Statement

Random Inputs ($w_0, w_1, \ldots, w_m$)



Output Quantity (y)

$$\boxed{y_n = f(w_0, w_1, \ldots, w_m)}$$

$$\boxed{\mathbb{P}(y_n \geq t) \leq ?}$$

Doob Sequence

$$\boxed{f(W_1, W_2, \ldots, W_n)}$$

$W_1, \ldots, W_n$ are independent

$$X_0 : \ \mathbb{E}(f(W_1, \ldots, W_n)) \longleftarrow \boxed{\text{Constant}}$$

$$X_1 : \ \mathbb{E}(f(W_1, W_2, \ldots, W_n) \mid W_1) \longleftarrow \boxed{\mathbb{E}(f(w_1, W_2, \ldots, W_n))}$$

$$X_2 : \ \mathbb{E}(f(W_1, W_2, \ldots, W_n) \mid W_1, W_2) \longleftarrow \boxed{\mathbb{E}(f(w_1, w_2, \ldots, W_n))}$$

$$\vdots$$

$$X_n : \ \mathbb{E}(f(W_1, W_2, \ldots, W_n) \mid W_1, \ldots, W_n) \longleftarrow \boxed{f(w_1, \ldots, w_n)}$$

Doob Sequences are Martingales

$$\begin{aligned}\mathbb{E}(X_{j+1} \mid W_j, \ldots, W_1) &= \mathbb{E}(\mathbb{E}(f(W_1, \ldots, W_n) \mid W_1, \ldots, W_{j+1}) \mid W_1, \ldots, W_j) \\ &= \mathbb{E}(f(W_1, \ldots, W_n) \mid W_1, \ldots, W_j) \\ &= X_j\end{aligned}$$

$$\boxed{\mathbb{E}(\mathbb{E}_B(X \mid A, B) \mid A) = \mathbb{E}(X \mid A)}$$

Method of Bounded Differences

$$\boxed{f(W_1, W_2, \ldots, W_n)}$$

$$\mathbb{P}\left(f(w_1, \ldots, w_n) \left\{ \begin{array}{l} \geq \mathbb{E}(f) + t \\ \leq \mathbb{E}(f) - t \end{array} \right. \right) \leq ??$$

$W_1, \ldots, W_n$ are independent

Lipschitz Condition:

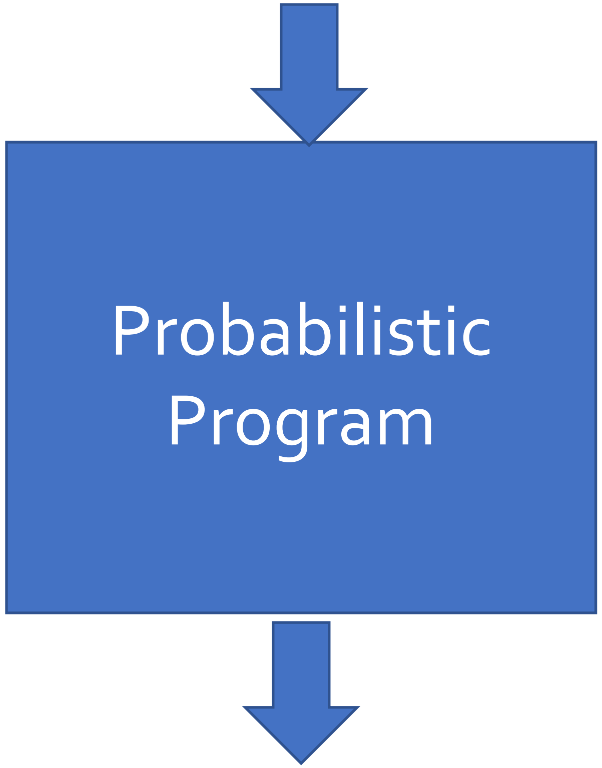
$$|f(w_1, \ldots, w_j, \ldots, w_n) - f(w_1, \ldots, \hat{w}_j, \ldots, w_n)| \leq c_j$$

Azuma Inequality Applied to Doob Martingale:

$$\mathbb{P}\left(f(w_1, \ldots, w_n) \left\{ \begin{array}{l} \geq \mathbb{E}(f) + t \\ \leq \mathbb{E}(f) - t \end{array} \right. \right) \leq \exp\left(\frac{-t^2}{2\sum_{j=1}^{n} c_j^2} \right)$$

Application to Programs

Random Inputs (w_0, w_1, … , w_m)



Output Quantity (y)

$$\boxed{y_n = f(w_0, w_1, \ldots, w_m)}$$

$$\boxed{\mathbb{P}(y_n \geq t) \leq ?}$$

1. Estimate Lipschitz bounds for each variable.
   - How? [Open Problem].

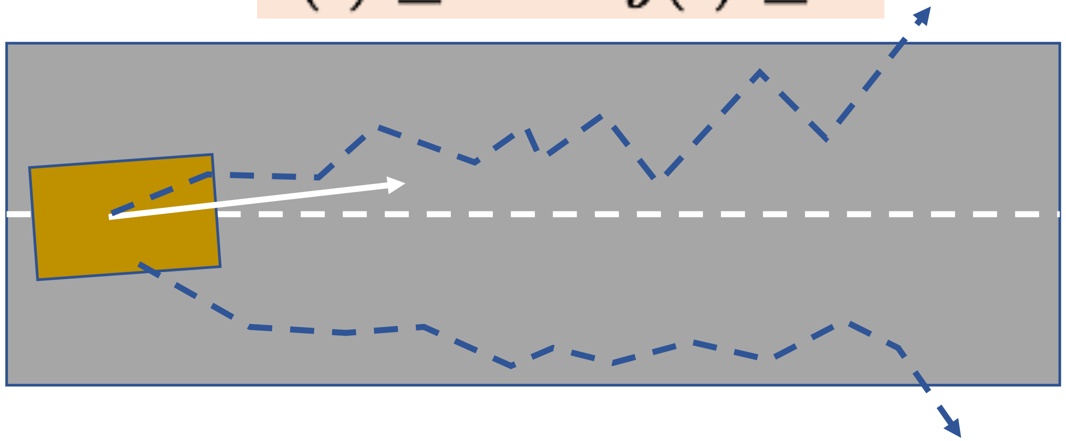
2. Apply Method of Bounded Differences.

Direct Application of Azuma’s Theorem

Concentration of Measure: Experiment #2

Vehicle on a road. (x, y, θ)

$$\theta(t) \geq 0 \ \wedge \ y(t) \geq L$$



$$\theta(t) \leq 0 \ \wedge \ y(t) \leq -L$$

$$y(t+1) = y(t) + 0.1\theta$$
$$\theta(t+1) = 0.99\theta(t) + w$$
$$w \in [-0.01, 0.01]$$
$$\mathbb{E}(w) = 0$$

$$\boxed{M(t): \ y(t) + 10\theta(t)}$$

$$\theta(t) \leq 0 \ \wedge \ y(t) \geq L \ \Rightarrow \ y(t) + 10\theta(t) \geq L$$
$$\theta(t) \leq 0 \ \wedge \ y(t) \leq -L \ \Rightarrow \ y(t) + 10\theta(t) \leq -L$$

Experiment #2: Azuma's Inequality

$$M(t): \; y(t) + 10\theta(t)$$

$$M(0): \; y(0) + 10\theta(0)$$

$$\mathbb{E}(M(t)) = \mathbb{E}(M(0)) = 0$$

Lipschitz Condition:

$$\begin{aligned} |M(t+1) - M(t)| &= |y(t+1) - y(t) + 10(\theta(t+1) - \theta(t))| \\ &= |0.1\theta(t) + 10(0.99\theta(t) + w(t+1) - \theta(t))| \\ &= |10w(t+1)| \\ &\leq 0.1 \end{aligned}$$

Experiment #2: Proving Bounds

$$\theta(t) \leq 0 \ \wedge \ y(t) \geq L \ \Rightarrow \ y(t) + 10\theta(t) \geq L$$

$$\mathbb{P}(M(t) \geq L) \leq \exp\left(\frac{-L^2}{0.02t}\right)$$

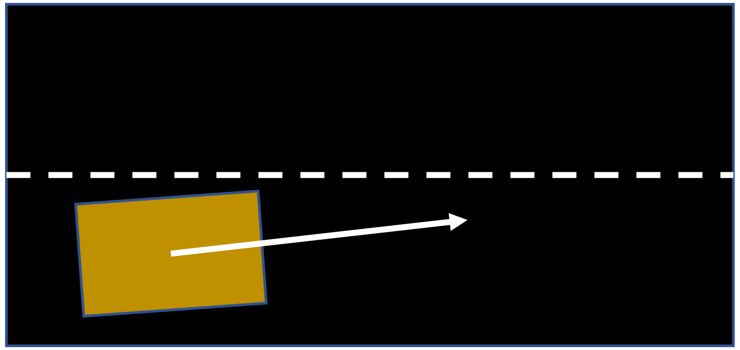
Fix t = 100

L	Azuma Inequality	Chernoff-Hoeffding
0.38	0.93	0.48
1.5	0.32	7.7×10^{-5}
3.0	0.011	9.5×10^{-14}
3.8	0.0073	3.8×10^{-19}

Automatic Inference of Martingales

Concentration of Measure: Experiment #2

Vehicle on a road. (x, y, θ)



$$y(t+1) = y(t) + 0.1\theta$$
$$\theta(t+1) = 0.99\theta(t) + w$$
$$w \in [-0.01, 0.01]$$
$$\mathbb{E}(w) = 0$$

$$\boxed{M(t):\ y(t) + 10\theta(t)}$$

How do we find martingales?

Super Martingales of Probabilistic Programs

$\mathbf{x}$

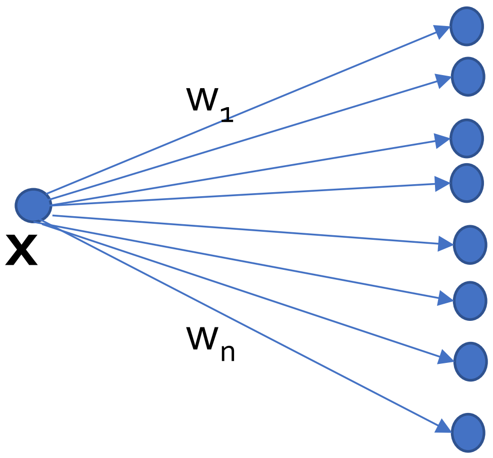
S

x := F(x, w)

$\mathbf{x}'$

Pre-Expectation of f w.r.t S

$$\mathsf{preE}(f, S) : \ \mathbb{E}(f(\mathbf{x}') \mid \mathbf{x})$$



Pre-Expectation Calculus [McIver & Morgan]

S_1

(x, y) := 2 * x + Uniform(-1, 2), - y + Uniform(-1, 1)

Pre-Expectation Example #1

(x, y) := 2 * x + Uniform(-1, 2), - y + Uniform(-1, 1)

$$\mathsf{preE}(x - y, S) = \mathbb{E}(x' - y' \mid x, y)$$

Pre-Expectation Example #2

```
if (x >= 0)
    x := x + Uniform(-1,2)
    y := y -1
else
    x := 2 * x – Uniform(-1, 1)
    y := y - 2
```

$$(x', y') = \begin{cases} (x + U(-1,2), y-1) & \text{if } x \geq 0 \\ (2x - U(-1,1), y-2) & \text{if } x < 0 \end{cases}$$

$$[\varphi] : \begin{cases} 1 & \text{if } \varphi \\ 0 & \text{otherwise} \end{cases}$$

$$\mathsf{preE}(x, S) = \mathbb{E}(x' | \ x, y)$$

Loop Supermartingales

```
var x1,.., xn
while (C)
  do
     S
  od
```

$f(x_1, \ldots, x_n)$ is a martingale expression iff

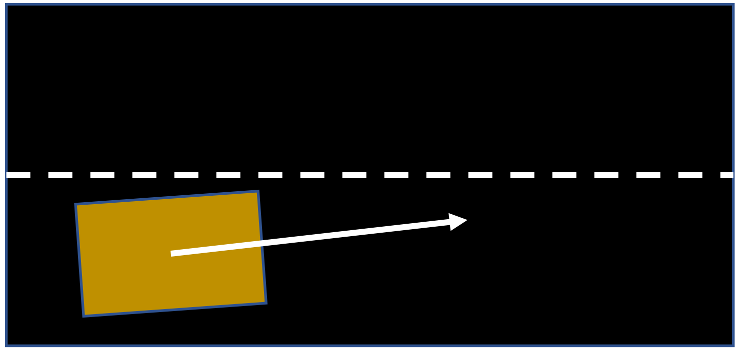
$$(\forall\ x_1, \ldots, x_n)\ \mathsf{preE}(f, S) = f(x_1, \ldots, x_n)$$

$g(x_1, \ldots, x_n)$ is a super martingale expression iff

$$(\forall\ x_1, \ldots, x_n)\ \mathsf{preE}(g, S) \leq g(x_1, \ldots, x_n)$$

Concentration of Measure: Experiment #2

Vehicle on a road. (x, y, θ)



$$y(t+1) = y(t) + 0.1\theta$$
$$\theta(t+1) = 0.99\theta(t) + w$$
$$w \in [-0.01, 0.01]$$
$$\mathbb{E}(w) = 0$$

```
while (true)
do                                   S
    y := y + 0.1 * th
   th := 0.99 th + randomW()
od
```

$$\boxed{M(t): \; y(t) + 10\theta(t)}$$

preE(y + 10 * th, S) = y + 10 * th

Automatic Inference of (Super) Martingale

[Katoen + McIver + Morgan, Gretz + Katoen, Chakarov + S]

1. Fix an unknown template form of the desired function.

$$c_1 y + c_2 \theta$$

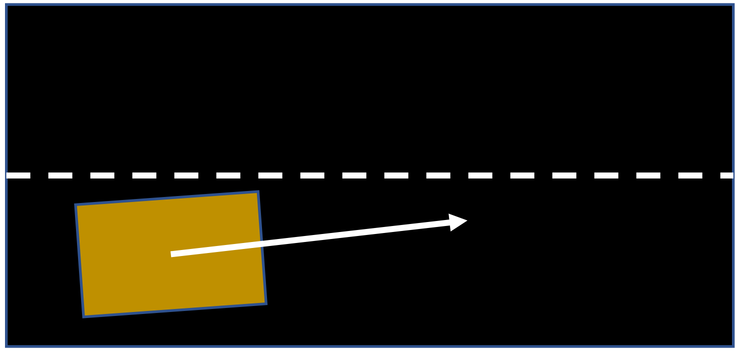
2. Use Farkas' Lemma (theorem of the alternative) to derive constraints [Colon+S+Sipma'03]

3. Solve to obtain (super) martingales.

$$c_1 : \ 1, \ c_2 : \ 10$$

Automatic Inference (Example)

Vehicle on a road. (x, y, θ)



$$x := x + 0.1(1 - \frac{1}{2}\theta^2)$$
$$y := y + 0.1\theta$$
$$\theta := 0.99\theta + w$$
$$\mathbb{E}(w) = 0$$

$$c_1 x^2 + c_2 y^2 + c_3 \theta^2 + c_4 \theta y + c_5 x + c_6 y + c_7 \theta + c_8 n$$

$2.985n + 150\theta^2 - 2.985x$	Martingale
$10\theta + y$	Martingale
$2000\theta y - 199n + 100y^2 + 1990x$	Martingale
$49n - 500x$	Supermartingale
$1000\theta - n$	Supermartingale
$10x - n$	Supermartingale
$-n - 1000\theta$	Supermartingale

Further Work on Martingale Inference #1

- Using Doob decomposition [Barthe et al. CAV 2016].

- Start from an given expression and iteratively derive a martingale.
  - Can derive very complex expressions.
  - Lots of avenues for future refinements here.

Further Work on Martingale Inference #2

- Exponential Supermartingales [Tedrake+Steinhardt' IJRR 2012]
  - Using Sum-of-Squares Inequalities and Semi-Definite Programming.
  - Clever tricks to avoid solving bilinear matrix inequalities.
  - Comparison with Azuma's inequality may be interesting.

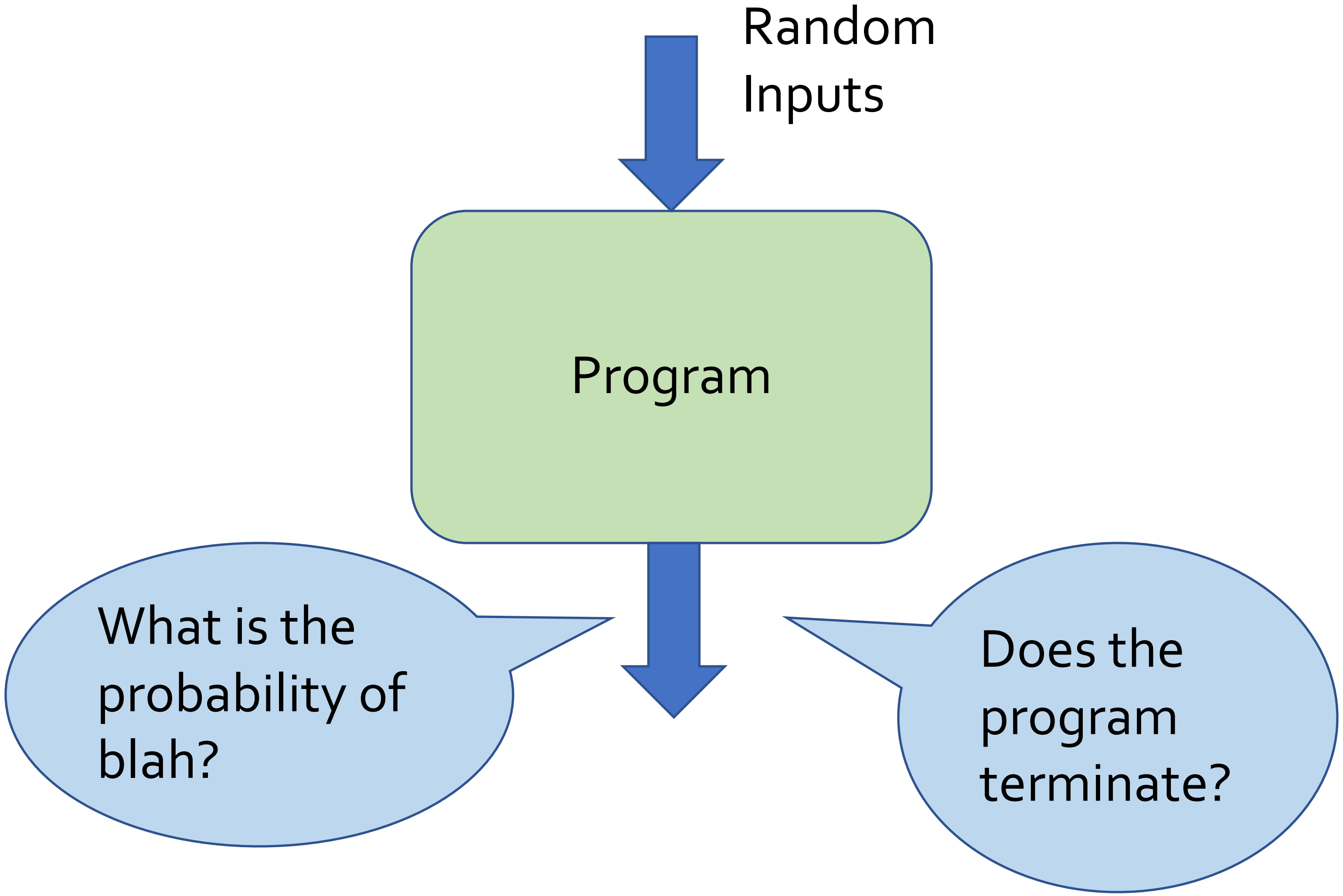
Probabilistic Program Analysis and Concentration of Measure

Part III: *Termination, Persistence and Recurrence, Almost Surely!*

Sriram Sankaranarayanan

University of Colorado, Boulder

Quantitative vs. Qualitative Questions



Qualitative Questions

- Almost Sure Termination/Reachability. $\boxed{\Diamond S}$
  - The program terminates *with probability 1*.
  - All executions eventually reach a desired set *with probability 1.*
- Almost Sure Persistence. $\boxed{\Diamond \Box S}$
  - The program executions reach a set S and remain in S forever.
- Almost Sure Recurrence $\boxed{\Box \Diamond S}$
  - The program executions visit S infinitely often.

Almost Sure Termination

```
while (x >= y)
    x := x + Uniform(-1, 1)
    y := y + Gaussian(1, 2.0)
```

Does this loop terminate?

Nonterminating execution

(10,8) → (11,8) → (12, 8) → (13, 8) → …

Almost Sure Termination.

Terminates with probability 1.

Measure of samples leading to non-termination is 0.

Proving Termination

```
while (x  >= y)
    x := x
    y := y  + 1
```

Ranking Function: x – y

- Decreases by 1 on each loop iteration.
- When negative, loop terminates.

```
while (x  >= y)
    x := x + U(-1,1)
    y := y  + N(1, 2)
```

Supermartingale Ranking Function: x – y

$$
\begin{aligned}
\mathsf{preE}(x - y, S) &= \mathbb{E}(x' - y' | x, y) \\
&= \mathbb{E}(x + U(-1,1) - y - N(1,2) | x, y) \\
&= x - y - 1
\end{aligned}
$$

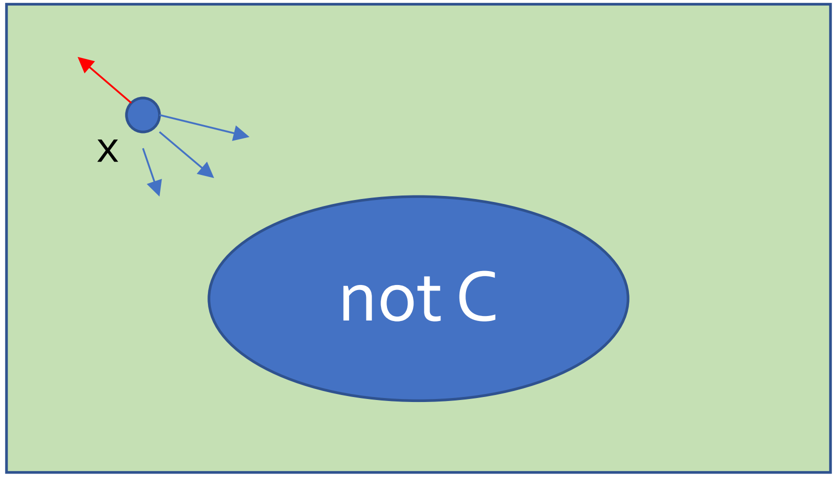
Supermartingale Ranking Functions (SMRF)

Function of program state: $f(x_1, \ldots, x_n)$

```
var x1, .., xn
while ( C ) do
      S
od
```

$$f \leq 0 \;\Rightarrow\; (\neg C)$$

$$\text{preE}(f, S) \leq f - \epsilon$$



- “Foster” Lyapunov Criteria (for discrete time Markov Chains).
- Ranking function analogues [McIver + Morgan]

Main Result

```
var x1, .., xn
while (C) do
      S
od
```

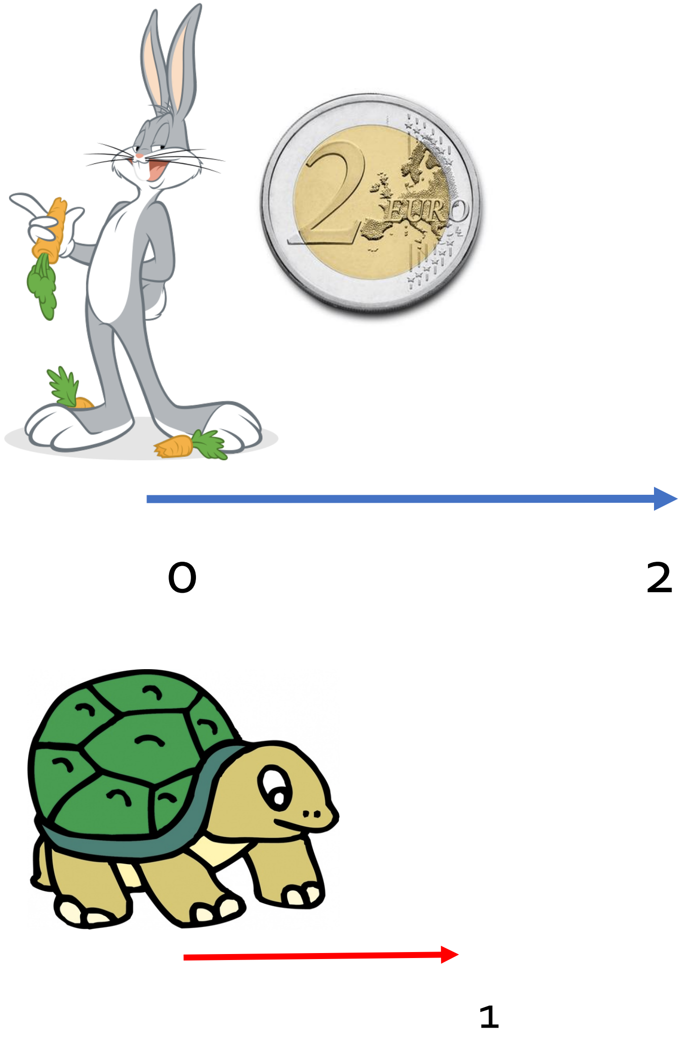
- Let $f(x_1, \ldots, x_n)$ be a SMRF.
- If f is positive over the initial state.
- Then f becomes negative almost surely upon repeatedly executing the loop body.

Corollary of Martingale Convergence Thm. (+ technicalities).

Example # 1

```
real h, t
// h is hare position
// t is tortoise position
while (t <= h)
    if (flip(0.5))
        h := h  + uniformRandom(0,2)
    t := t + 1
// Almost sure termination?
```

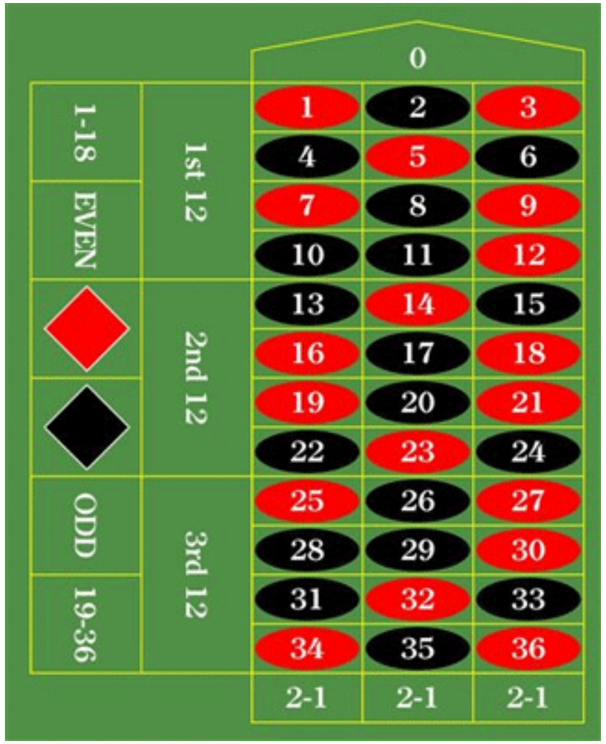
$$\boxed{h - t}$$



"Slow and steady wins the race *almost surely*"

Example #2 : Betting Strategy For Roulette

```
i := 0;
money := 10,  bet
while (money >= 10 ) {
      bet := rand(5,10)
       money := money - bet
       if (flip(36/37)) // bank lost
            if flip(1/3)  //  col. 1
               if flip(1/2)
                   money := money + 1.6*bet // Red
               else
                   money := money + 1.2*bet // Black
             elseif flip(1/2) //  col. 2
                    if flip(1/3)
                         money := money + 1.6*bet; // Red
                    else  money := money + 1.2*bet // Black
             else  //  col. 3
                     if flip(2/3)
                          money := money + 0.4*bet // Red
           i := i + 1 }
```



money – 10 is a SMRF

Obtaining Completeness

- SMRFs are not complete for proving termination.

```
x = 0
while (x != 1 and x != -1)
    if (flip(0.5))
        x := x + 1
    else
        x := x - 1
// Almost sure termination
```

The program can be shown to terminate almost surely.

No SMRF exists.

Completeness assuming the time taken to terminate (stopping time) is integrable
[Fioriti, Hermanns et al.'15].
Proving bounds on time taken to terminate. [Chatterjee et al.'16, Kaminski et al'16].
Complexity of proving almost sure termination. [Kaminski + Katoen '15].

A note of caution…

while C do
    S

(not C) holds?

```
x := 0
while ( x  ! = 1)
    if ( flip(0.5))
        x := x + 1
    else
        x := x - 1
```

x = 1 holds

x is a martingale of the program

E(x) = 0 at initial state.
E(x) = 0 after each loop iteration.

E(x) = 0 holds when program terminates?

Facts about expected values at each loop iteration
are not necessarily true when the program terminates.

Doob's Optional Stopping Theorem: Provides condition when we can transfer.
[Fioriti, Hermanns POPL'15].

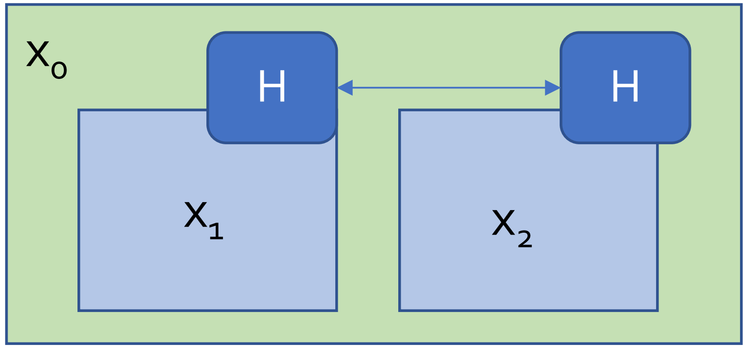
Persistence (and Recurrence)

Beyond Termination..

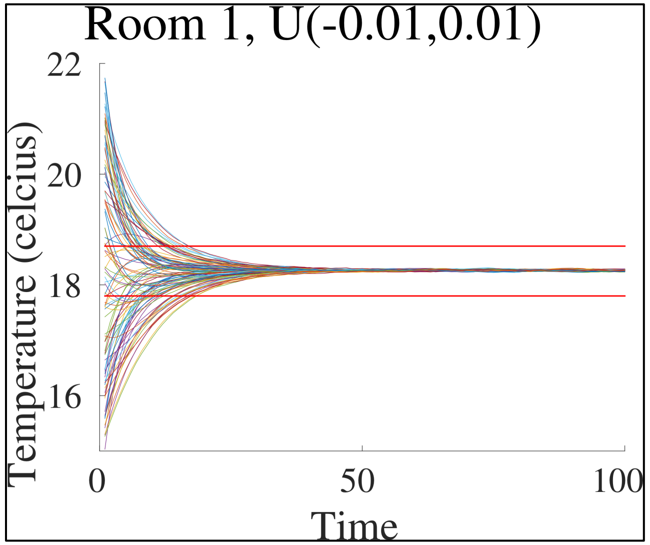
- We are often interested in proving more complex temporal properties.

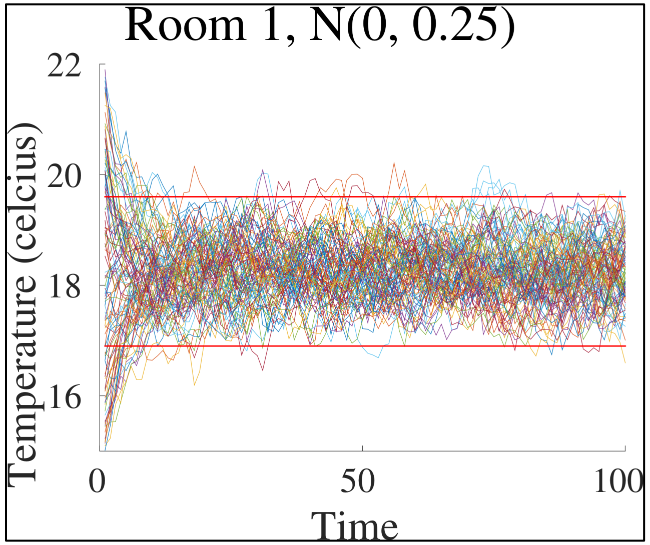
- Two papers in the same conference!
  - [Chakarov+Voronin+S' TACAS 16]
  - [Dimitrova+Fioriti+Hermanns+Majumdar'TACAS 16]
  - Both based on ideas using martingale theory.

Room Heater Example [Abate et al. 2010]

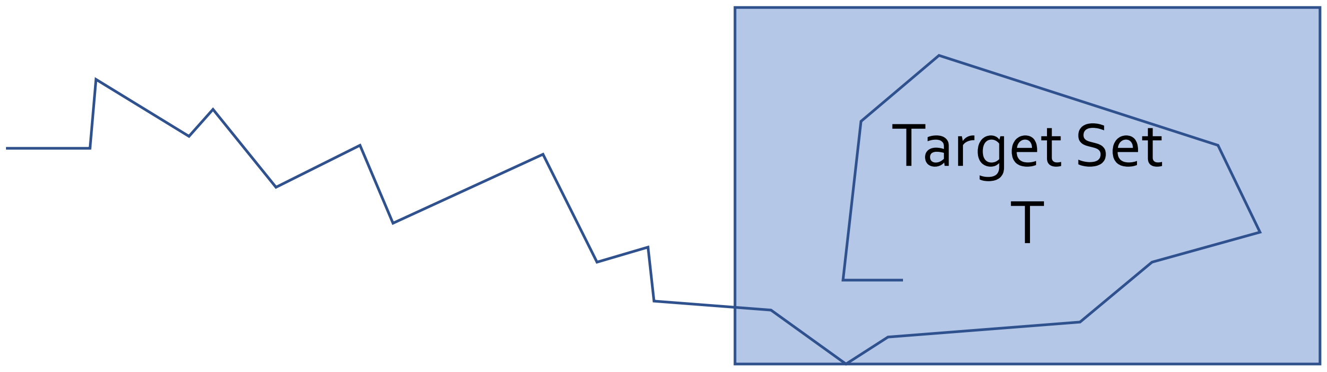


$$x_1' = x_1 + b_1(x_0 - x_1) + a(x_2 - x_1) + c_1(1 - \sigma(x_1)) + \nu_1$$
$$x_2' = x_2 + b_2(x_0 - x_2) + a(x_1 - x_2) + c_2(1 - \sigma(x_2)) + \nu_2$$



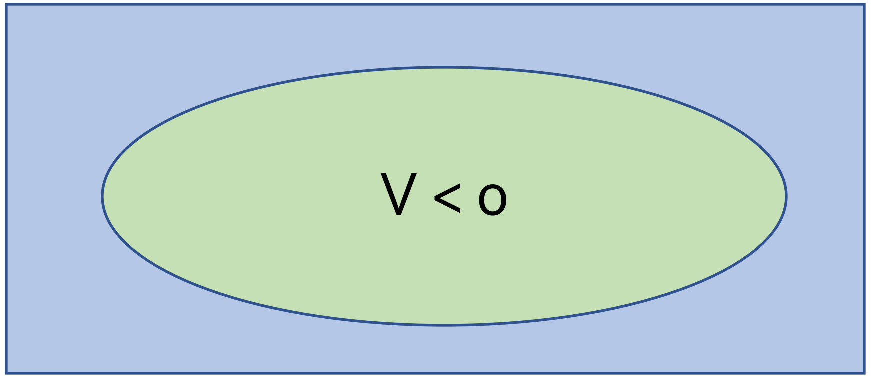


Persistence



Almost surely, all behaviors enter T eventually and stay in there forever.

From Termination to Persistence

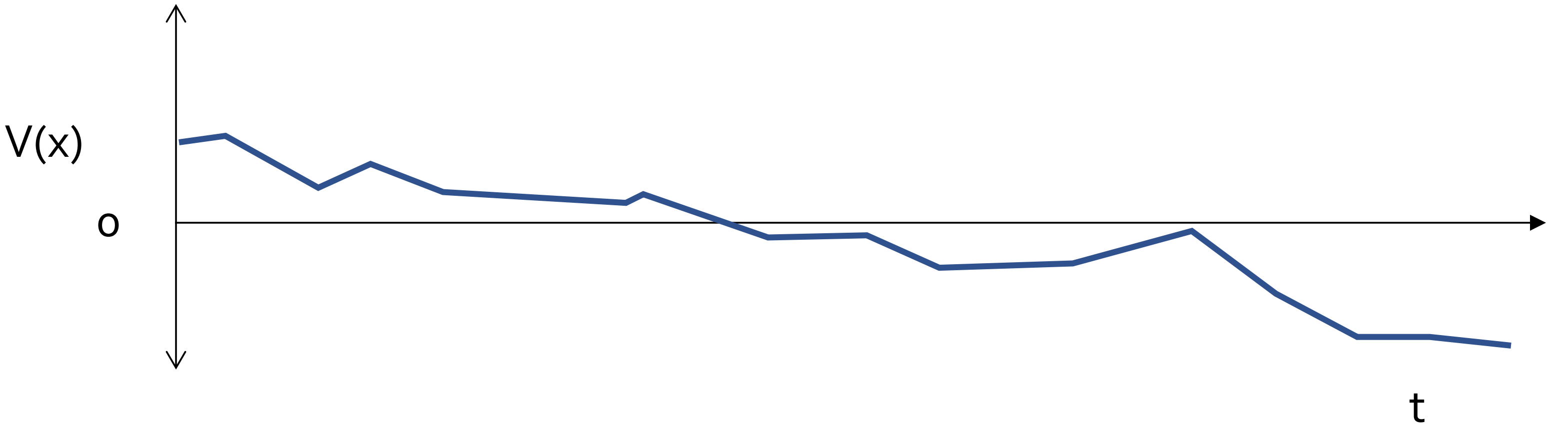


Target Set

$$V(\mathbf{x}) < 0 \;\Rightarrow\; \mathbf{x} \in T$$

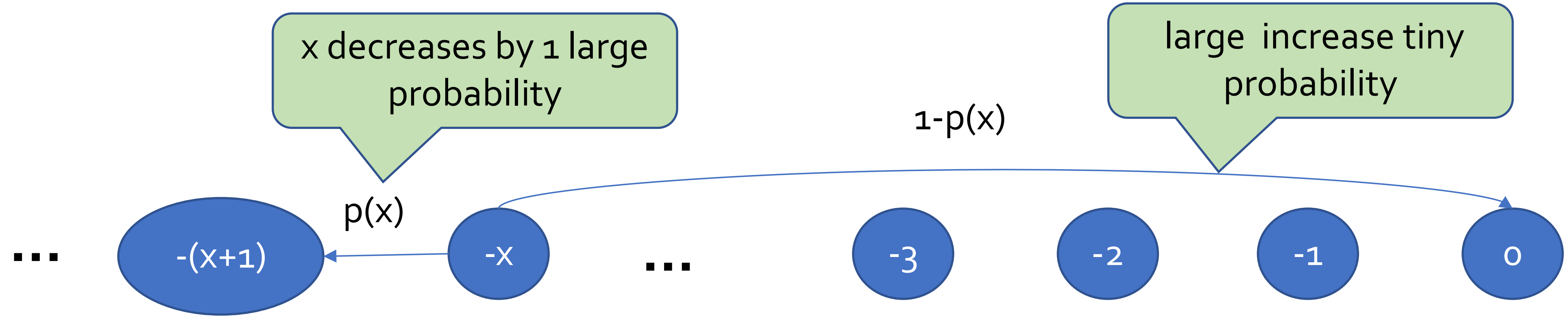
Decrease Rule

$$\mathbb{E}(V(\mathbf{x}')|\mathbf{x}) \leq V(\mathbf{x}) - \epsilon$$



Unsound!

Unsoundness of SMRFs for Persistence



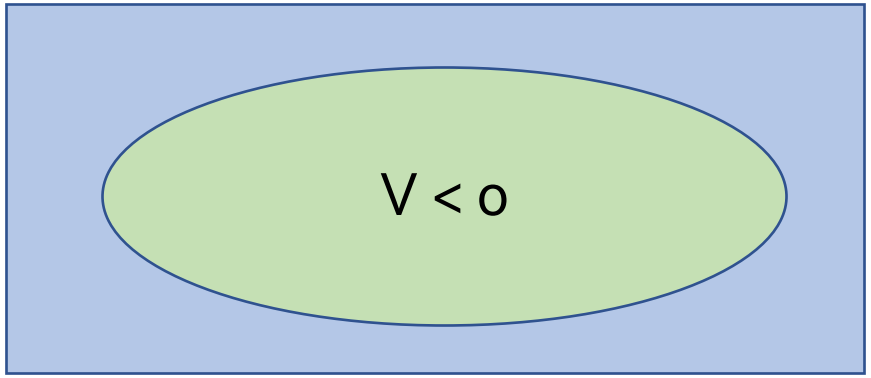
$$p(x) = 1 - \frac{0.5}{1-x}$$

V(x) = x satisfies the conditions for SMRF.

The chain visits 0 infinitely often almost surely!

$$\sum_{x=-\infty}^{-1} 1 - p(x) = \sum_{x=1}^{\infty} \frac{0.5}{x+1}$$
$$= \infty$$

Bounded Increase Condition



Target Set

$$V(\mathbf{x}) < 0 \ \Rightarrow \ \mathbf{x} \in T$$

Decrease Rule

$$\mathbb{E}(V(\mathbf{x}')|\mathbf{x}) \leq V(\mathbf{x}) - \epsilon$$

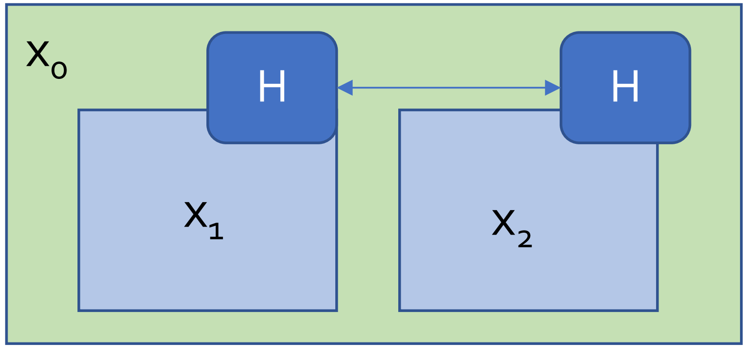
Bounded Decrease Condition

$$V(\mathbf{x}') - V(\mathbf{x}) \leq M$$

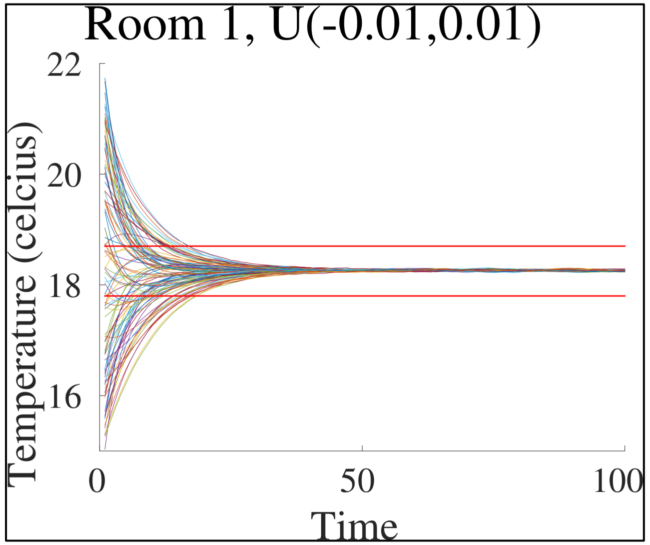
If V satisfies conditions above, then

$\Diamond\Box(T)$ almost surely

Room Heater Example [Abate et al. 2010]



$$x_1' = x_1 + b_1(x_0 - x_1) + a(x_2 - x_1) + c_1(1 - \sigma(x_1)) + \nu_1$$
$$x_2' = x_2 + b_2(x_0 - x_2) + a(x_1 - x_2) + c_2(1 - \sigma(x_2)) + \nu_2$$



Using Sum-Of-Squares Programming

$$(x_1 - 18.3)^2 + (x_2 - 18.8)^2$$

Open Directions

Challenge #1: Symbolic Domains

- Incorporate Booleans, Graphs and other domains.

- Common in randomized algorithms.

- Benefit by careful mechanization.

- Application areas:
  - Dynamics on graphs and social networks.
  - Graph rewriting systems (Graph Grammars).
  - Self-assembling systems.

Challenge #2: Concentration of Measure Inequalities

- Understanding when concentration of measure inequalities work.
  - Using more properties about the underlying distributions.
- Designer Inequalities.
  - Symbolic inference of property specific inequalities.

Thank You!

Thank you to University of Minho and the Organizers of the Summer School.

Work supported by US NSF under award # CCF-1320069 (primary source of support), CNS-0953941, CNS-1016994 and CPS-1035845. All opinions expressed are those of the authors and not necessarily of the NSF