# Static Analysis of Programs with Probabilities

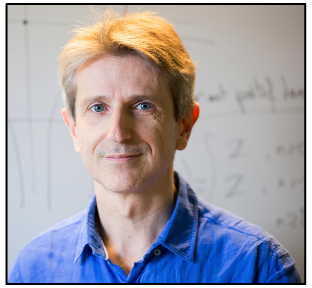
Sriram Sankaranarayanan

University of Colorado, Boulder, USA.

# Joint Work

Aleksandar Chakarov
Univ. Colorado, Boulder
now at Phase Change

Olivier Bouissou
CEA, now at Mathworks

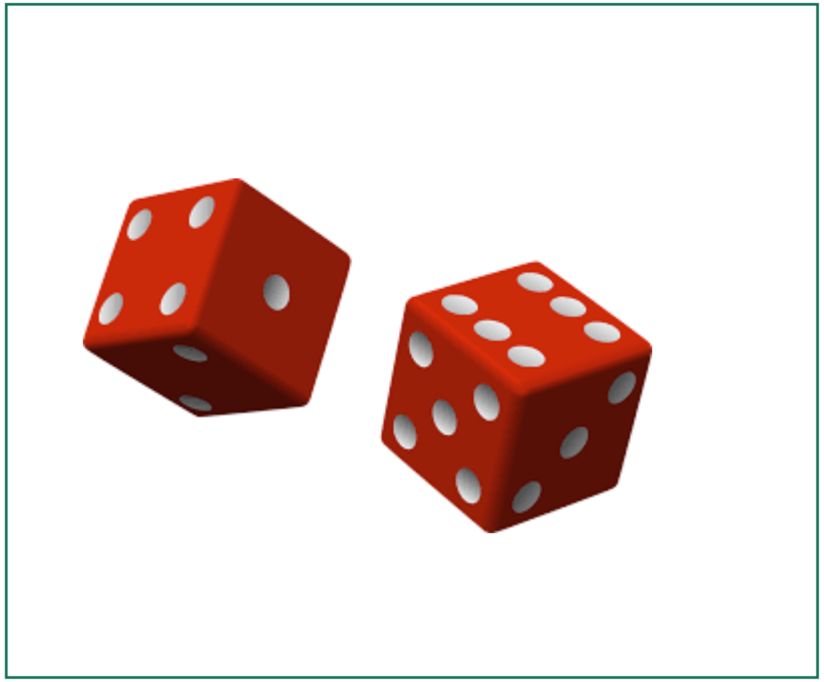


Eric Goubault
Ecole Polytechnique

Sylvie Putot
Ecole Polytechnique

Yuen-Lam Voronin
Univ. Colorado, Boulder

# What is this talk about?



VERSUS

Stochastic
Randomized

Demonic
Worst-Case

# Programs with Probabilities

```
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)
```

Probabilistic Statements

Probability Estimate

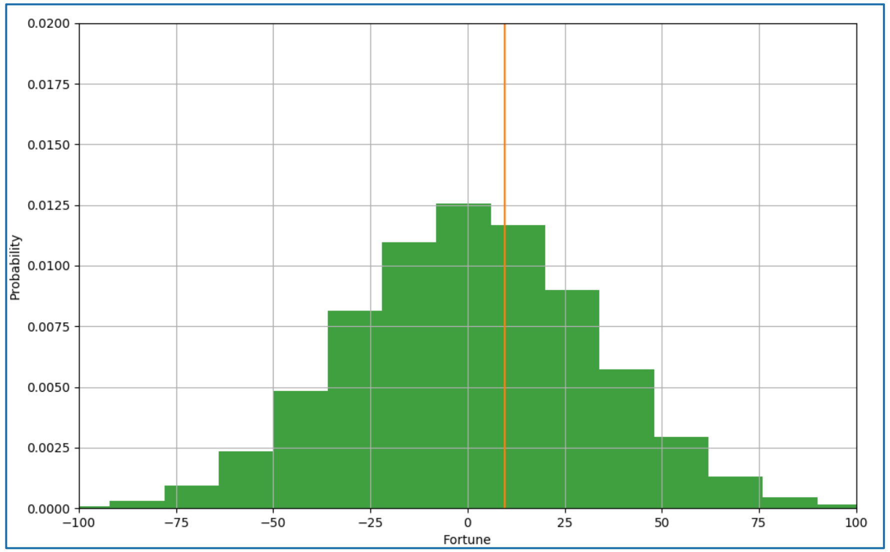
# Example #1: Coin Toss

Heads → Gain one dollar

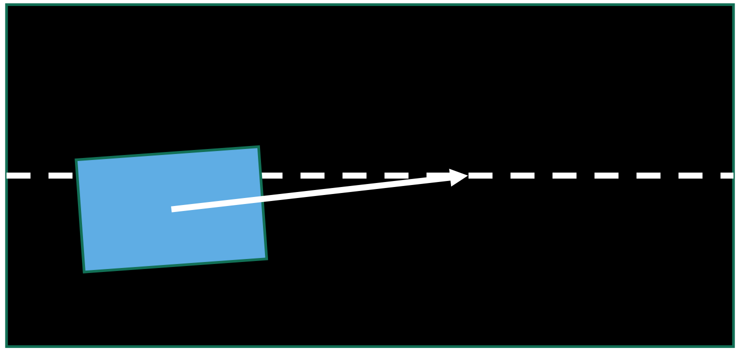
Repeat 1000 times.

Tails → Lose one dollar

```
fortune := 1000
repeat(1000)
    if flip(0.5):
        fortune := fortune +1
    else:
        fortune := fortune -1
assert fortune >= 0
```

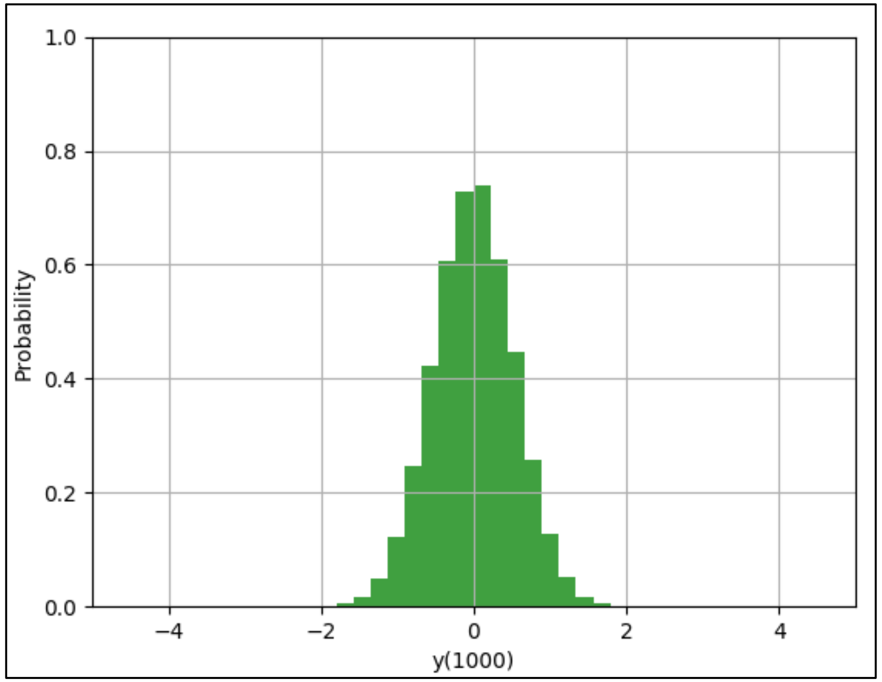


# Example #2: Vehicle on a road

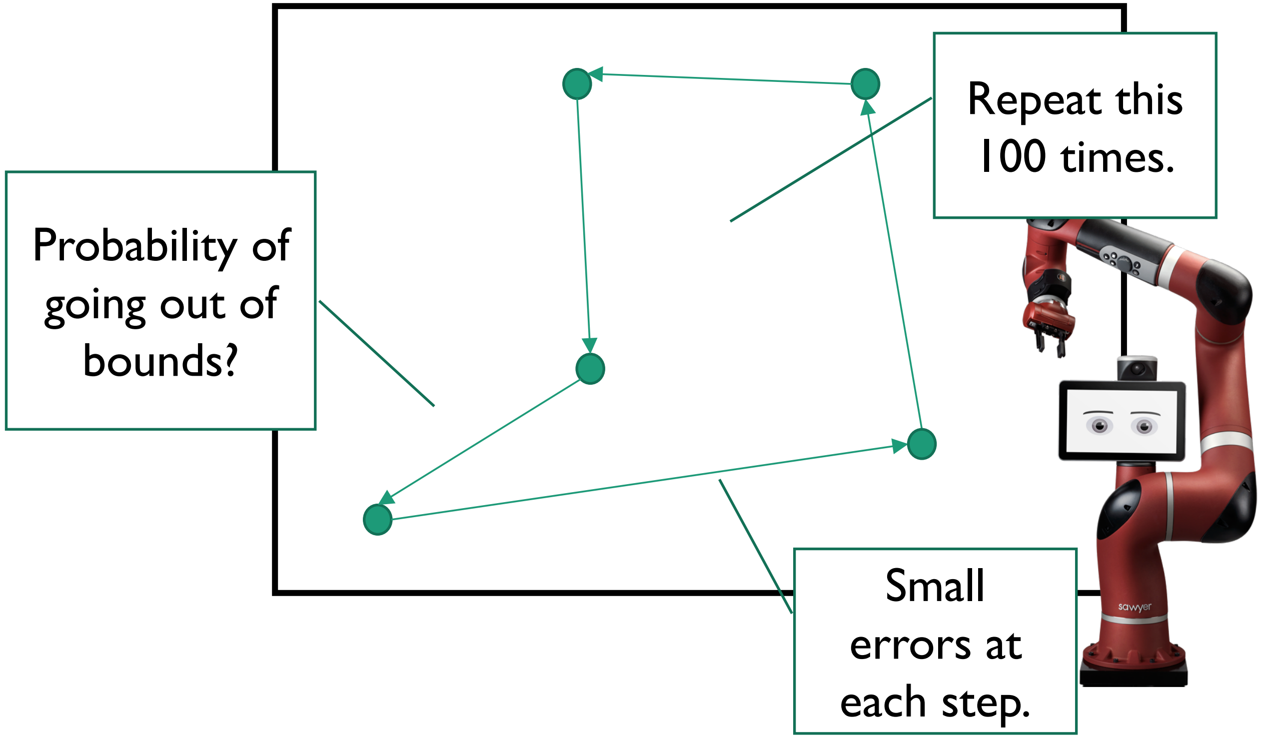


```
y = 0, theta = 0, x = 0
repeat(1000)
   x := x + 0.1 * cos(theta)
   y := y + 0.1 * sin(theta)
   theta := 0.8 * theta + Normal(0, 0.1)
assert (y <= 5.0)
```

$$x(t+1) = x(t) + 0.1\cos(\theta)$$
$$y(t+1) = y(t) + 0.1\sin(\theta)$$
$$\theta(t+1) = 0.8\theta(t) + w$$
$$w \sim \mathcal{N}(0, 0.1)$$



# Example #3: Repetitive Robot



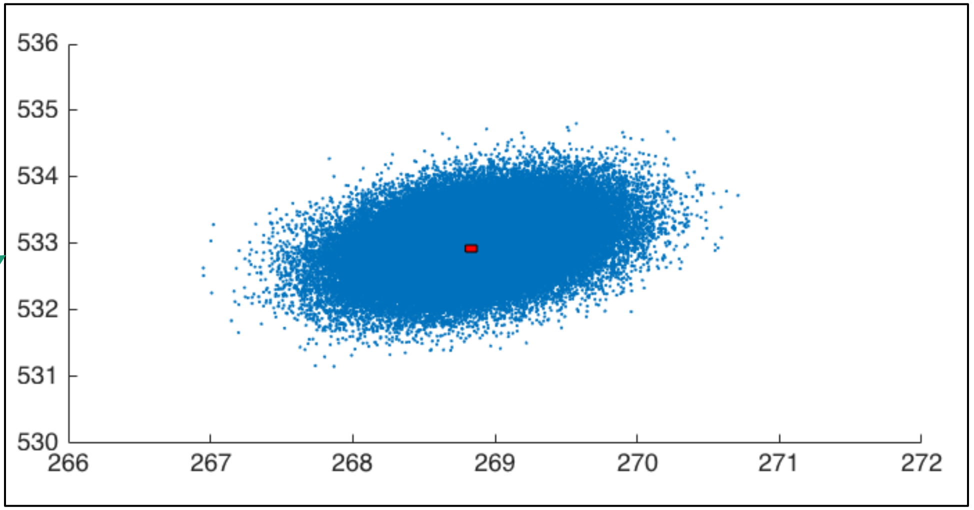

Sawyer Robotic Arm
(rethink robotics)

```
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)
```

# Repetitive Robot

```
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 10^5 Simulations




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

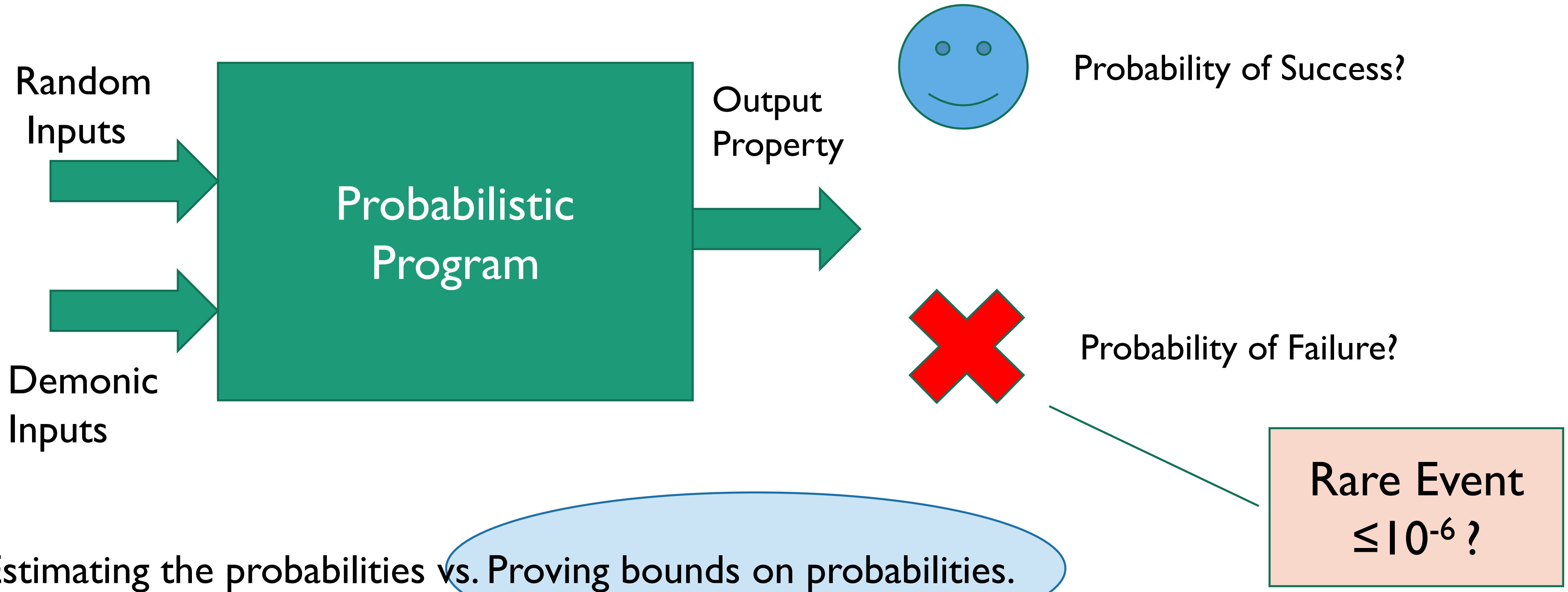
# Systems Acting Under Disturbances

Stochastic Verification
Reliability
Stochastic Controls
Uncertainty Quantification
AI

External Disturbances → System → Output → Property → Yes / No

"Classic" Formal Verification.
"Set-Valued" Robust Control.

# Reasoning about Uncertainty




Estimating the probabilities vs. Proving bounds on probabilities.

# Static Analysis of Probabilities

# Semantics

```
real x,y,z;
initially x is Normal(0,1),
           y is Uniform(-1,1);
initially z is Uniform(0,10);
while (true)
  if (z <= 10)
     x := x +1 + 2*Normal(0,1);
     y :=  y – 2 + NONDET (0,1) ;
     z := z + 1;
  else
     x := x +1;
     y := y – 2;
     z := z -1;
```

is a

Markov Process

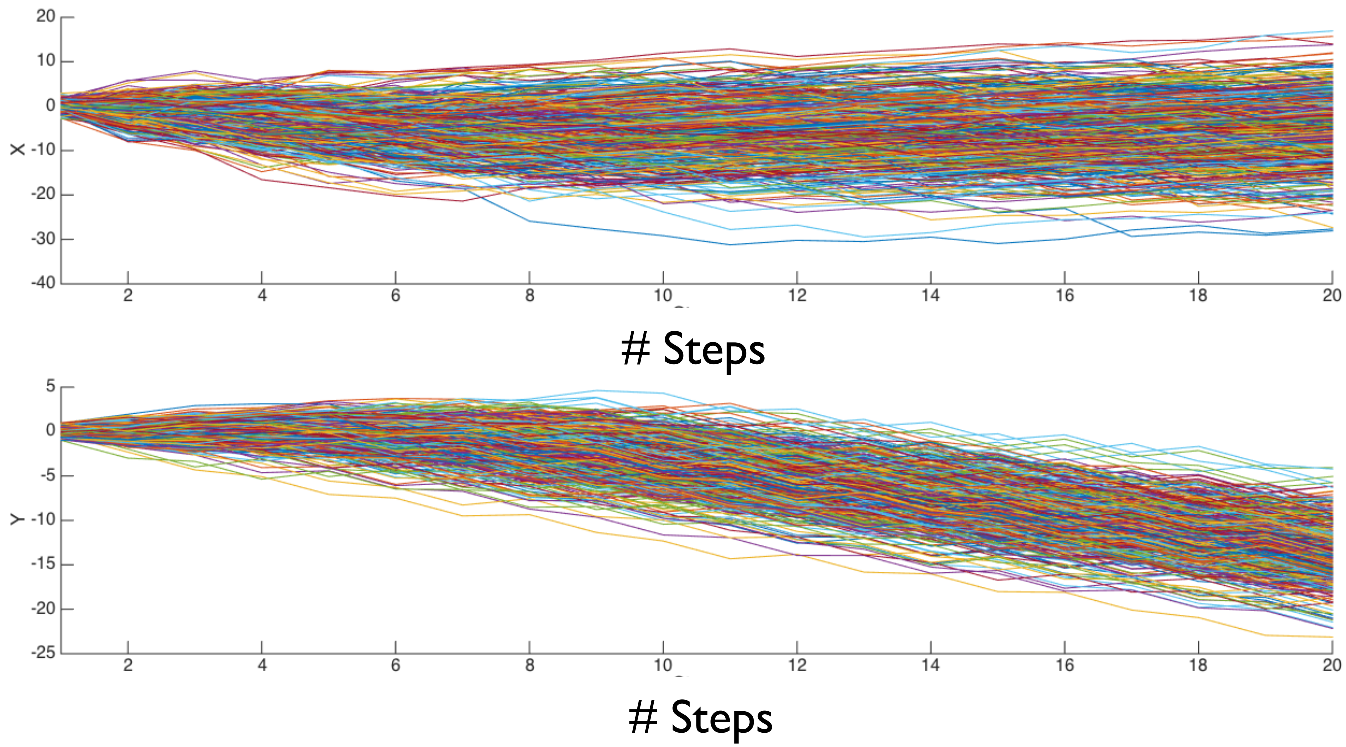
Complicated Semantics

Skip for this talk

# Sample Path Semantics

[Kozen'1981]

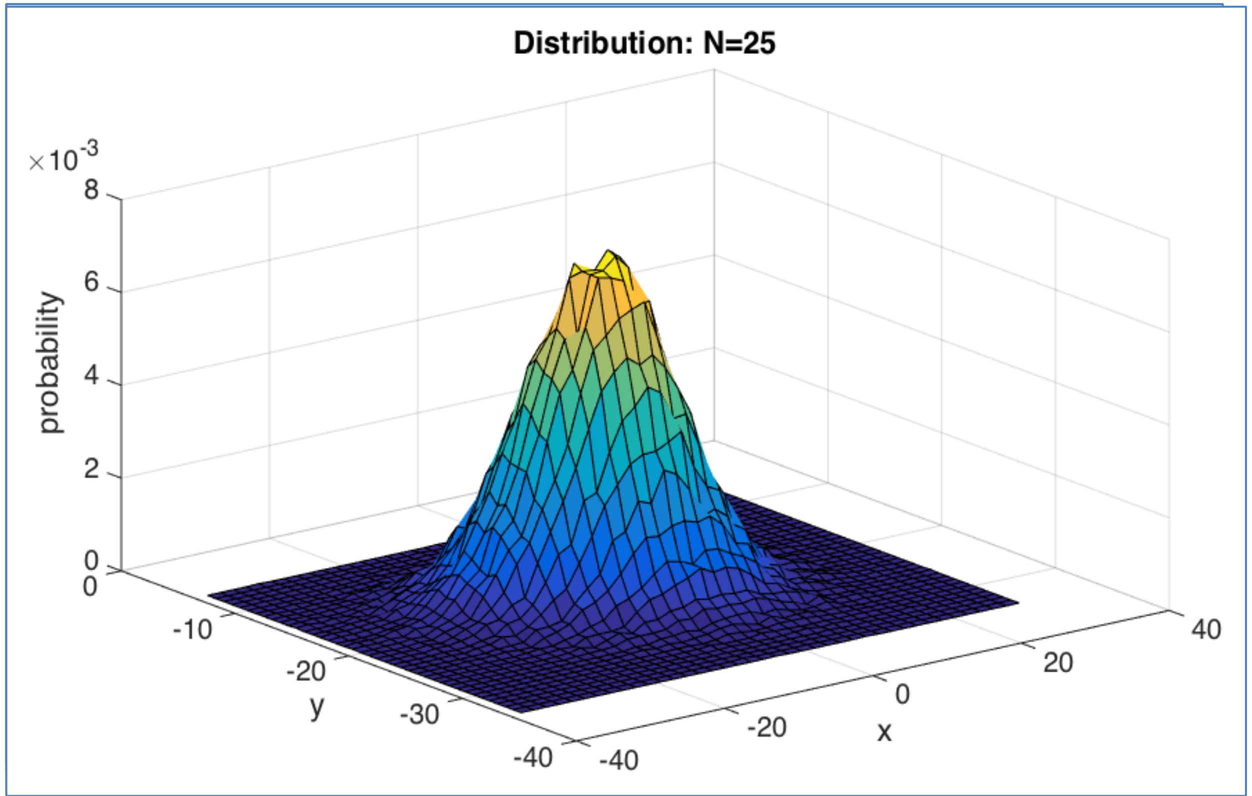
"Sample path" semantics.




```
real x,y,z;
initially x is Normal(0,1),
          y is Uniform(-1,1);
initially z is Uniform(0,10);
while (true)
  if (z <= 10)
     x := x -1 + 2*Normal(0,1);
     y := y − 2 + Uniform(-1,1);
     z := z + 1;
  else
     x := x +1;
     y := y − 2;
     z := z -1;
```

# Distribution Transformer Semantics

[Kozen'1981]



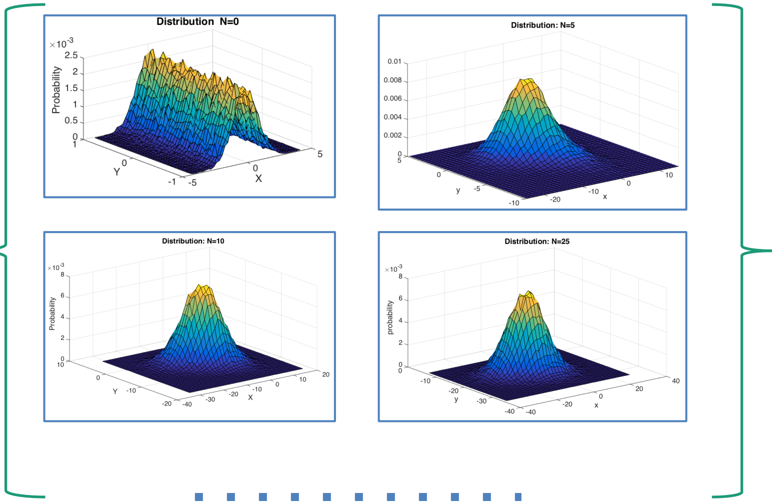

```
real x,y,z;
initially x is Normal(0,1),
          y is Uniform(-1,1);
initially z is Uniform(0,10);
while (true)
  if (z <= 10)
     x := x -1 + 2*Normal(0,1);
     y :=  y – 2 + Uniform(-1,1);
     z := z + 1;
  else
     x := x +1;
     y := y – 2;
     z := z -1;
```

# Comparison with "Classical" Programs

| **"Classical" Programs** | **Probabilistic Programs** |
| --- | --- |
| State (x:10, y:25, z:15) | Distributions x: N(0,1), y: U(-1,1), z: Poisson(5) |
| Sets of States | Sets of Distributions |
| Abstract Domains | Probabilistic Abstract Domains |

# Reachable Set of Distributions

```
real x, y;
real z;
initially x is Normal(0,1),
          y is Uniform(-1,1);
initially z is Uniform(0,10);
while (true)
  if (z <= 10)
     x := x -1 + 2*Normal(0,1);
     y :=  y – 2 + Uniform(-1,1);
     z := z + 1;
   else
     x := x +1;
     y := y – 2;
     z := z -1;
```

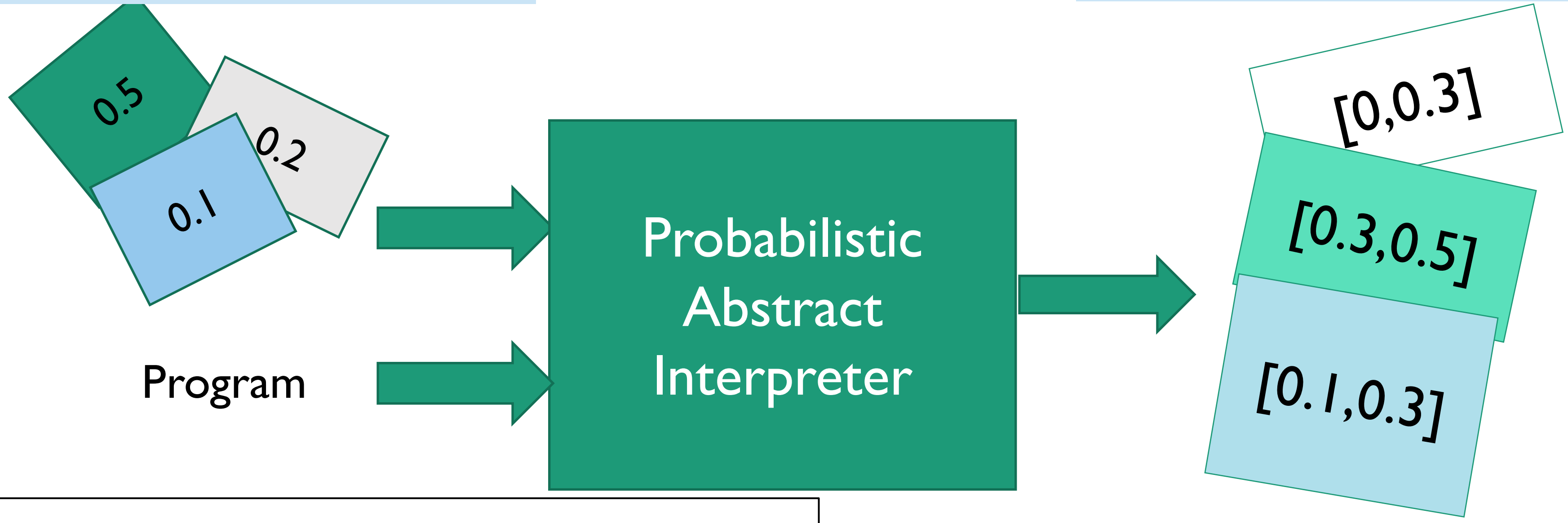


# Probabilistic Abstract Interpretation

[Monniaux, Cousot+Monerau, Mardziel + Hicks, Bouissou+Goubault+Putot, **S**+Chakarov+Gulwani, …]

Abstraction of Initial Distribution

Abstraction of reachable distributions



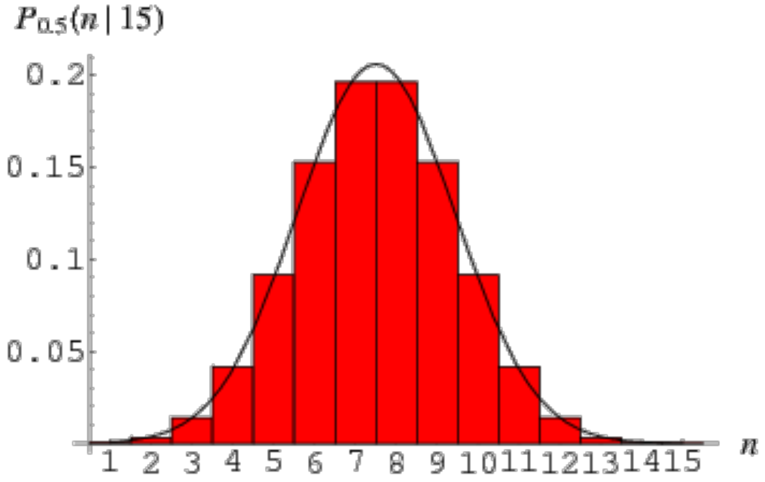

How to:
1. Systematically abstract distributions?
2. Propagate abstract distributions through programs?
3. Answer queries on the results?

Probability of x <= 135?

[0.956, 0.989]

# Approach #1: Discretization

[Monniaux, Mardziel+Hicks,Cousot+Monerau]




Partition domain into cells.
Associate range of probability with each cell.

Systematically abstract distributions?

Propagate abstract distributions through programs?

Use Standard Forwards/Backwards Abstract Interpretation (with modifications)

Answer queries on the results?

"Discrete" Integration
Volume Computation *(expensive)*

# Discretization

- Tradeoff: precise bounds vs number of cells.
- Off-the-shelf use of abstract interpretation tools.
- Conceptually easy to handle nondeterminism + stochastic choices.

➢ Does not scale to large number of random variables.

➢ Loops may require widening → precision loss.

# Approach #2: Probabilistic Calculii

[Bouissou+Goubault+Putot,
Bouissou+ Goubault + Putot+ Chakarov+S]

- How do 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) <= ??
```

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

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

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

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

# Probabilitic Affine Forms

Systematically abstract distributions?

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

$$w_i \in [a_i, b_i]$$
$$\mathbb{E}(w_i) \in [c_i, d_i]$$
$$\mathbb{E}(w_i^2) \in [\ell_i, u_i]$$
$$\mathbb{E}(w_i w_j) \in [f_{ij}, g_{ij}]$$

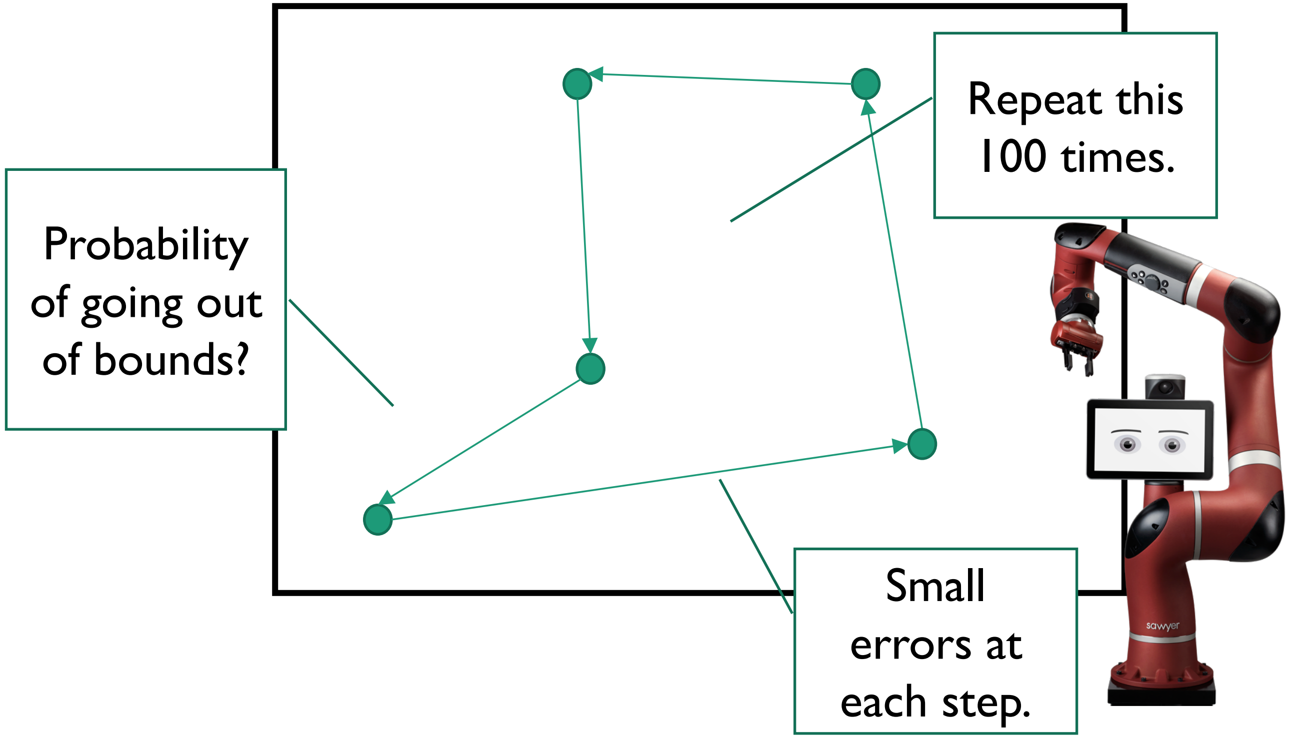
Propagate abstract distributions through programs?

Modified Affine Form Calculus
Conditional Branches

Answer queries on the results?

Volume Computation *(expensive)*
Concentration of Measure Inequalities
*(cheap but not fully general)*

# 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)
```

# Repetitive Robot: Affine Form

$$x : \left( \begin{array}{c} [268.78, 268.82] + w_1 + [0.984, 0.985] w_2 \\ +[0.030, 0.031] w_3 - w_4 \\ +[0.030, 0.031] w_5 - w_6 \\ +[0.49, 0.51] w_7 + [0.90, 0.91] w_8 + \\ -w_9 + [0.90, 0.91] w_{10} + \\ \cdots \\ [0.03, 0.031] w_{6892} - w_{6893} + \\ w_{6896} - w_{6898} - w_{6899} \end{array} \right).$$

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

[Bouissou+Chakaraov+Goubault+Putot+S'TACAS 2016]

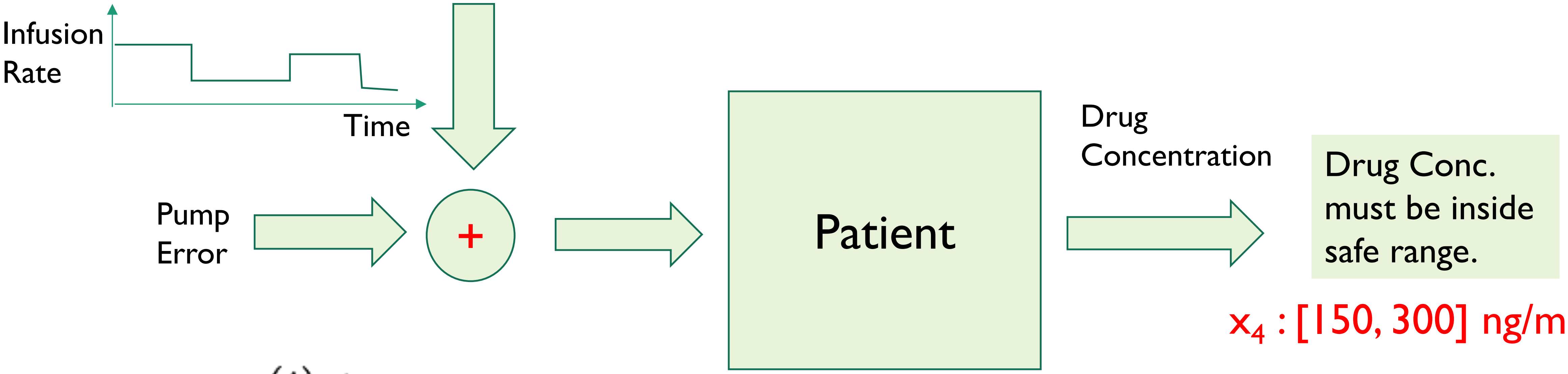
# Repetitive Robot (Cont.)

Bounds computation using Chernoff-Hoeffding Inequality:

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

# Anesthesia (Fentanyl) Infusion

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




$x_4$ : [150, 300] ng/ml

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

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

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

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

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

# 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
```

[Bouissou+Chakaraov+Goubault+Putot+**S**'TACAS 2016]

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

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

$$\mathbb{P}(x_4 \leq 300\mathtt{ng/ml}) \leq 7 \times 10^{-13}$$
$$\mathbb{P}(x_4 \geq 150\mathtt{ng/ml}) \leq 10^{-23}$$

# Affine Form-Based Approach

- ✓ Generalizes to nonlinear computation
  - ✓ Polynomials, Trigonometric Functions, Hyperbolic Functions.
- ✓ Relation to polynomial chaos approximations [Xiu+Karandiakis]
  - ✓ Wiener-Askey Approximation Scheme.
- ➢ Conditional Branches.
  - ➢ Current Solution: discretize domain of the affine form into smaller boxes.
- ➢ Unbounded Loops.

# Approach #3: Deductive

[McIver+Morgan+Katoen,
Chakarov+**S**, Chatterjee et al.,
Fioriti et al.]

Systematically abstract distributions?

```
real x,y,z
initially x is Normal(0,1),
          y is Uniform(-1,1),
          z is Uniform(0,10);
while (true)
      if (z < 10)
            x := x -1 + 2*Normal(0,1);
            y :=  y – 2 + Uniform(-1,1);
            z := z + 1;
      else
            x := x +1;
            y := y – 2;
            z := z -1;
```

$$\left\{\begin{array}{rcl} \mathbb{E}(x+z) & = & 5 \\ \mathbb{E}(y) & = & 2n \\ \mathbb{E}(z) & \leq & 11 \\ \mathbb{E}(z) & \leq & 5+n \\ \mathbb{E}(z) & \geq & 5-n \end{array}\right\}$$

Facts about the moments of distributions.
Loop Invariants.

# Deducing Properties of Distributions

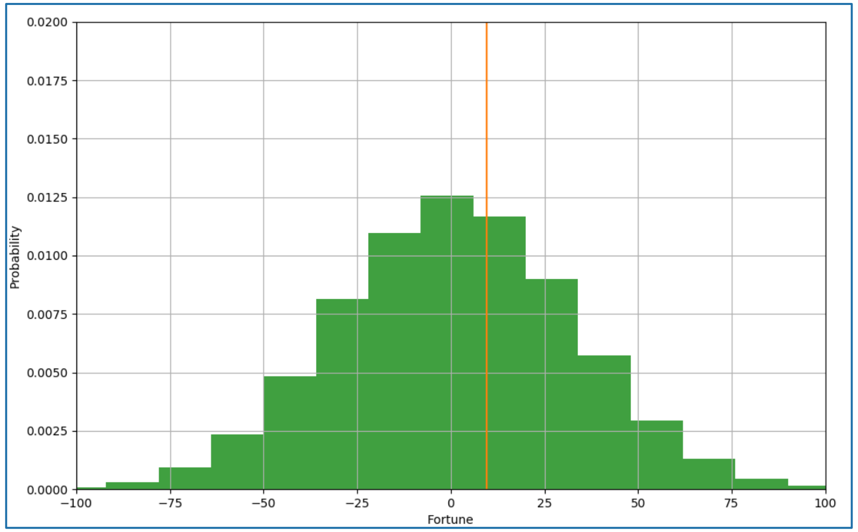
- Early work by McIver and Morgan.
  - Pre-Expectation calculus for programs with probabilities.
  - Restricted to finite domain random variables.
- Generalizing McIver and Morgan's work [Chakarov + S ' CAV 2013].
  - Connections with Supermartingales.
  - Handle continuous random variables.
  - Concentration of Measure Inequalities.

# Coin Tossing Example

Heads → Gain one dollar

Repeat N times.

Tails → Lose one dollar

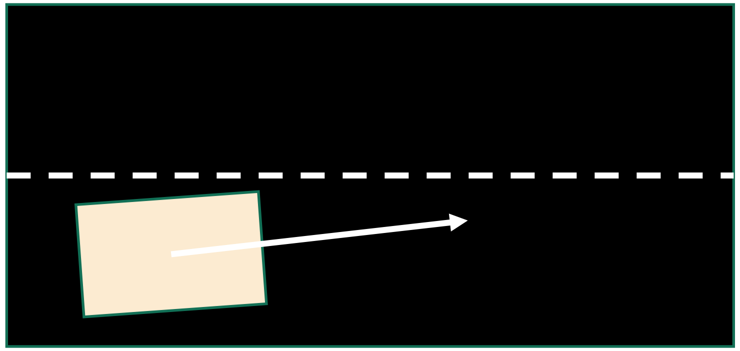



X Dollars → Toss Coin Once → ? Dollars

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

Expected fortune in next step = fortune in current step.

# Vehicle on the Road



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

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

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

Expected value in next step = value in current step.

# Martingale

Martingale is a special kind of stochastic process.

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

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

# Super/SubMartingales

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

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

# Super Martingales and Loop Invariants

```
real x,y,z
initially x is Normal(0,1),
          y is Uniform(-1,1),
          z is Uniform(0,10);
while (true)
    if (z < 10)
        x := x + 1 + 2*Normal(0,1);
        y :=  y – 2 + Uniform(-1,1);
        z := z + 1;
    else
        x := x +1;
        y := y – 2;
        z := z -1;
```

2 * x + y is a Martingale

$$(\forall\ x, y, z)\ \mathbb{E}(2x' + y' | x, y, z) = 2x + y$$

$$\boxed{\mathbb{E}(2x + y) = 0}$$

# Automatic Inference of (Super) Martingale

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

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

$$c_1 y + c_2 \theta$$

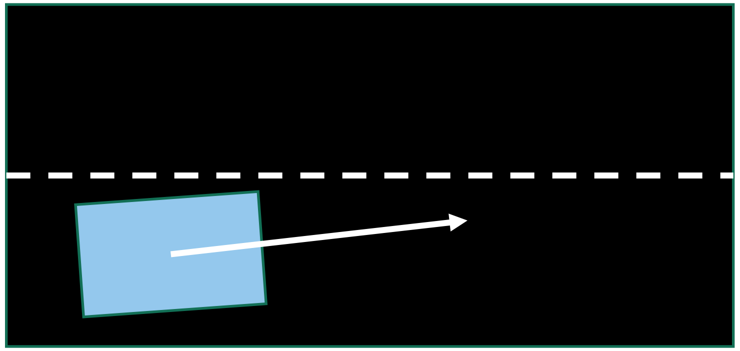
2. Use Farkas' Lemma to derive constraints [Colon+S+Sipma'03]

3. Solve to obtain (super) martingales.

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

# Automatic Inference (Example)

Vehicle on a road. $(x, y, \theta)$



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

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

| | |
|---|---|
| $2.985n + 150\theta^2 - 2.985x$ | 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 |

How do we use super martingales to answer queries?

# Azuma's Inequality for Martingales

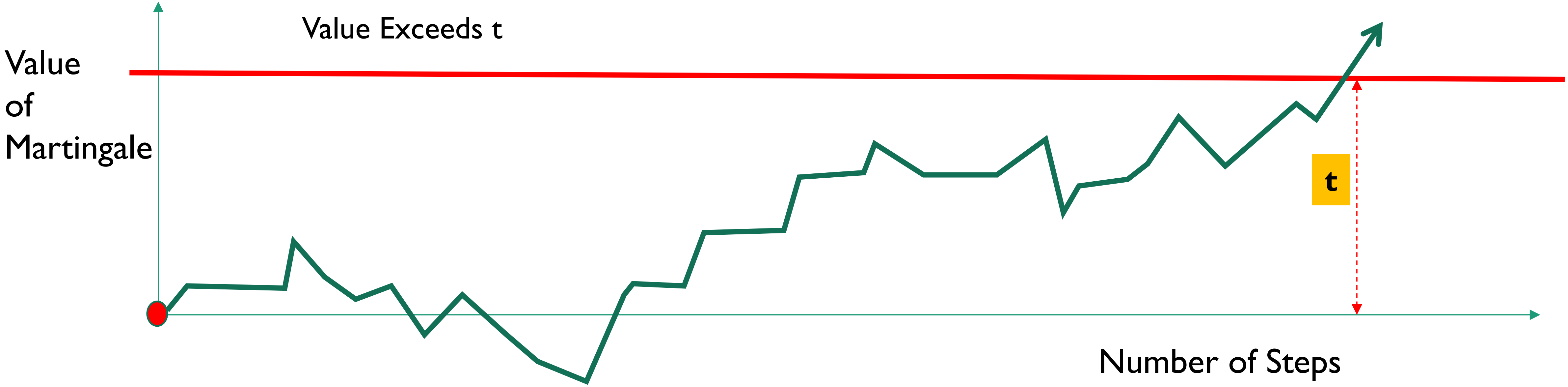
$X_0, \ldots, X_n$ stochastic process.

$|X_i - X_{i-1}| \leq c_i, \quad i > 0$ Lipschitz Condition

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

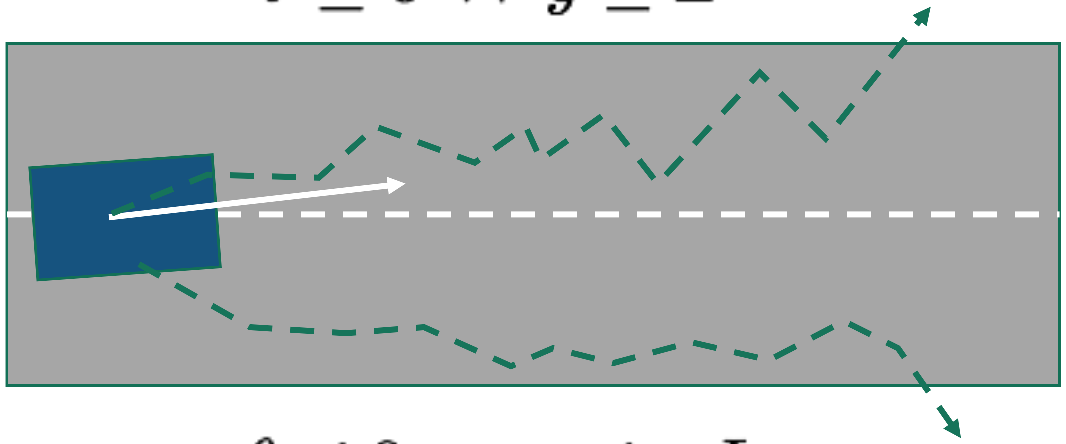
# Azuma Inequality (pictorially)




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

# Example: Vehicle on the Road

$\theta \geq 0 \ \wedge \ y \geq L$



$\theta \leq 0 \ \wedge \ y \leq -L$

$$M = y + 10\theta$$

$$\theta \geq 0 \ \wedge \ y \geq L \ \models \ y + 10\theta \geq L$$

$$\theta \leq 0 \ \wedge \ y \leq -L \ \models \ y + 10\theta \leq -L$$

# Experiment #2: Proving Bounds

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

Fix j = 100 steps (~ 10 seconds)

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

# Beyond Supermartingales

[McIver+Morgan+Katoen, Chakarov+**S**, Chatterjee et al., Fioriti et al.]

Systematically abstract distributions?

```
real x,y,z
initially x is Normal(0,1),
          y is Uniform(-1,1),
          z is Uniform(0,10);
while (true)
      if (z < 10)
            x := x -1 + 2*Normal(0,1);
            y := y - 2 + Uniform(-1,1);
            z := z + 1;
      else
            x := x +1;
            y := y - 2;
            z := z -1;
```

$$\begin{array}{rcl} \mathbb{E}(x+z) & = & 5 \\ \mathbb{E}(y) & = & 2n \\ \mathbb{E}(z) & \leq & 11 \\ \mathbb{E}(z) & \leq & 5+n \\ \mathbb{E}(z) & \geq & 5-n \end{array}$$

SuperMartingales
``Singly-Inductive'' Invariants

# Inductive Expectation Invariants

[Chakarov+**S**' SAS 2014]

Polyhedron:

$$A\mathbf{x} \leq \mathbf{b}$$

x: State.
Set of States

Polyhedron over measures:

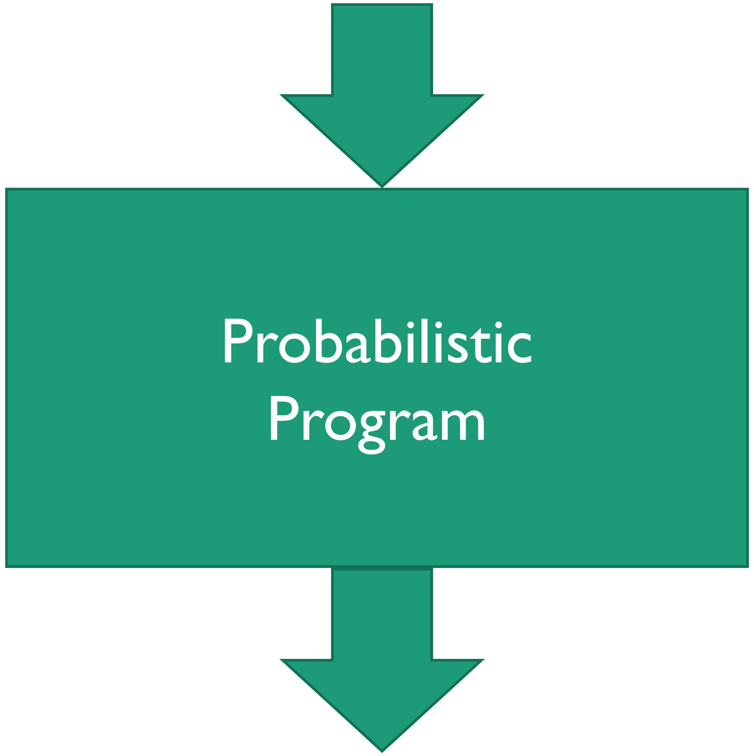
$$\mathbb{E}(A\mathbf{x}) \leq \mathbf{b}$$

x: Measure.
Set of Measures

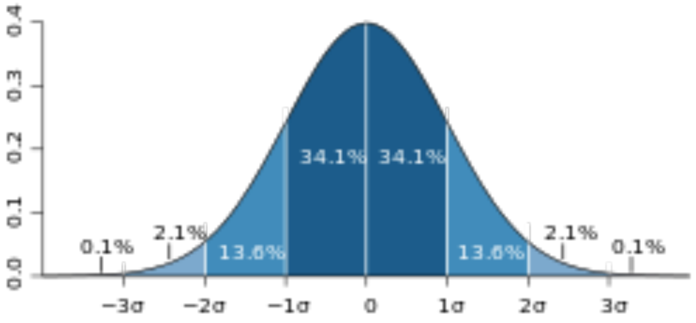
# Open Challenges

# Challenge # 1: Conditioning/Observations

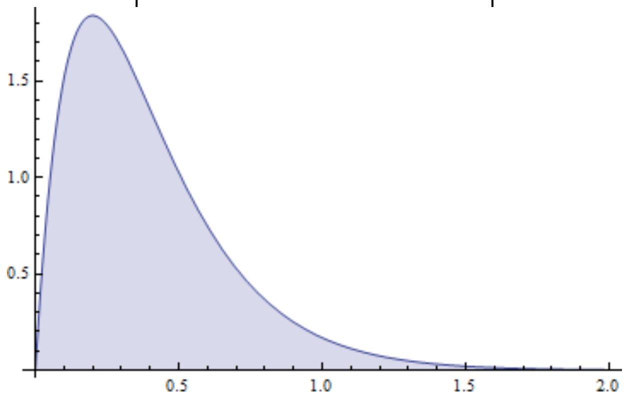
Parameters




Measurements/Observations




Prior Distribution




Posterior Distribution

# Conditioning/Observations

```
theta ~ Uniform[0,1]
tails := false
count = 0
while (not tails):
    tails := flip(theta)
    count := count  + 1
observe(count == 25);
assert(theta >= 0.6)
```

Applications
- Machine Learning.
- Filtering/State Estimation/Sensor Fusion.
- Data Driven Modeling.

Semantics of conditioning is *very tricky.*
*[Heunen et al. LICS 2017]*

# Challenge #2: Scalable Analysis

Uncertainty reasoning for large programs.

- Biological Systems
- Protein Folding
- Large Cyber-Physical Systems.

# Challenge #3: 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.

# Thank You

This work was supported by the US
National Science Foundation (NSF) under
Award # 1320069 and # 1646556.

All opinions expressed  are those of the authors
 and not necessarily of the NSF.