Absynth*

Bounded Expectations: Resource Analysis for Probabilistic Programs

Van Chan Ngo | Carnegie Mellon University

Dagstuhl Seminar 17291

*the name originally comes from Tom Reps. **Absynth** is built based on **Patis**

Overview

- Static analysis for deriving bounds on the expected resource usage of probabilistic programs
- Fully automated using off-the-shelf LP solving
- Multivariate polynomial bounds on the inputs
- Prototype implementation and benchmark design*

*available upon request at channgo@cmu.edu

Simple Programs

Current position

Boundary

```
int x, n;
x = 0;
while (x < n) {
  x = x + 1
  [3/4]
  x = x - 1;
  tick(1);
}
```

Cost 1 time unit

Simulates a random walk that ends when the walker passes the boundary

Each time unit:

- Goes forward 1 step with p = 3/4
- Goes backward 1 step with p = 1/4

What is the expected value of elapsed time?

Simple Programs

Current position

Boundary

```
int x, n;
x = 0;
while (x < n) {
    x = x + 1
    [3/4]
    x = x - 1;
    tick(1);
}
```

Cost 1 time unit

Simulates a random walk that ends when the walker passes the boundary

Each time unit:

- Goes fo th p = 3/4
- Goes ba ith p = 1/4

2max(0,n - x)

What is the expected value of elapsed time?

Trapped Miner

0: safety, 1: otherwise

number of independent sending

```
int flag, n;
while (n > 0) {
   { flag = 1;
   while (flag > 0) {
      { flag = 0; tick(3) }
      [1/3]
      { {flag = 1; tick(5)}
        [1/2]
        {flag = 1; tick(7);} }
   } }
   [1/2]
   skip;
   n = n - 1;
}
```

number of hours in channel

A miner is sent to a mine n time independently

With p = 1/2, the miner is trapped

When being trapped, 3 doors to open

- Door 1: takes 3 hours to safety
- Door 2: takes 5 hours to the mine
- Door 3: takes 7 hours to the mine

What is the expected time to reach safety?

Trapped Miner

0: safety, 1: otherwise

number of independent sending

```
int flag, n;
while (n > 0) {
   { flag = 1;
   while (flag > 0) {
      { flag = 0; tick(3) }
      [1/3]
      { {flag = 1; tick(5)}
        [1/2]
        {flag = 1; tick(7);} }
   } }
   [1/2]
   skip;
   n = n - 1;
}
```

number of hours in channel

A miner is sent to a mine n time independently

With p = 1/2, the miner is trapped

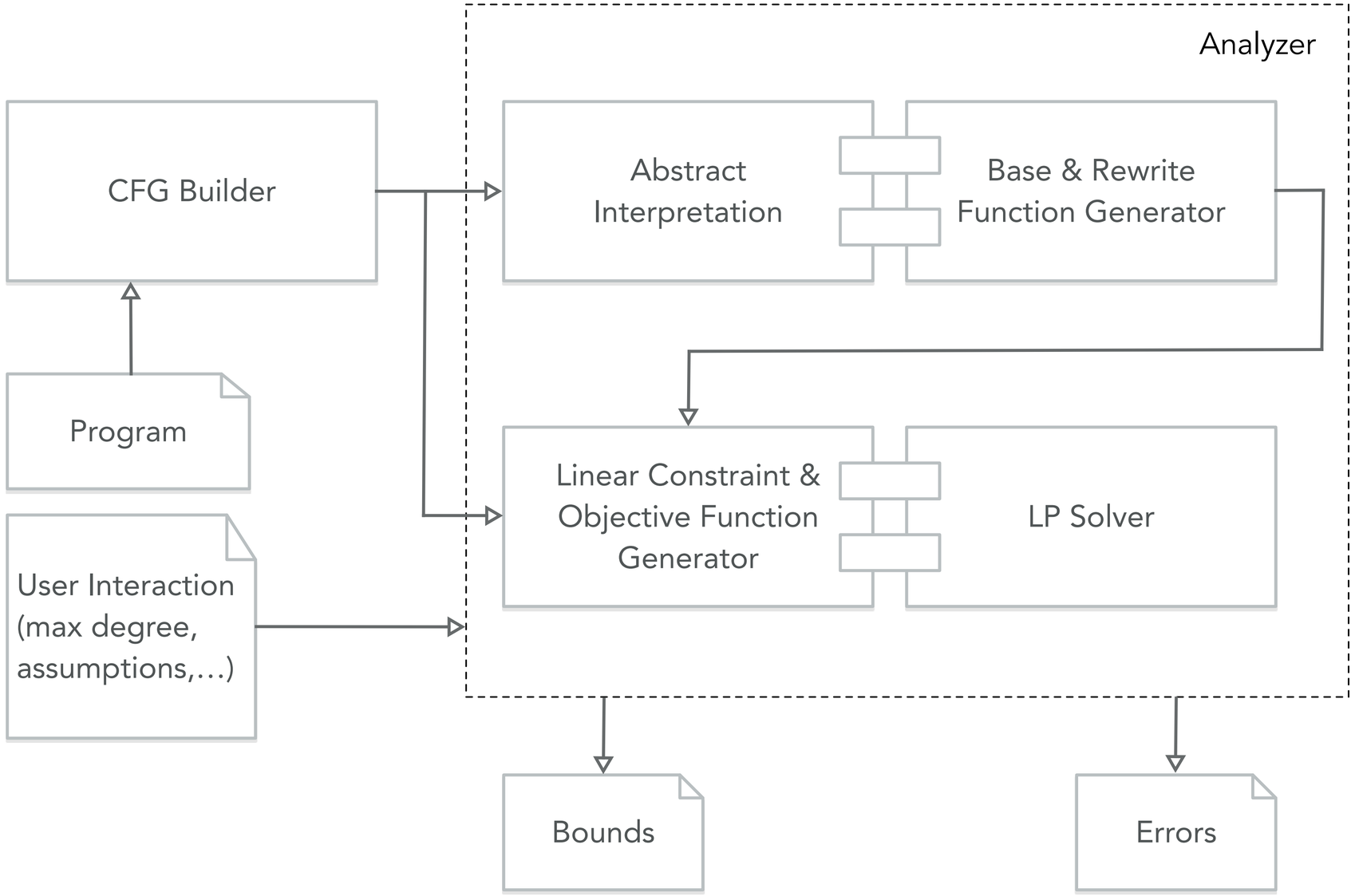
When being trapped, 3 doors to open

- Door 1: takes 3 hours to safety
- Door 2: the mine
- Door 3: the mine

7.5max(0,n)

What is the expected time to reach safety?

Architecture



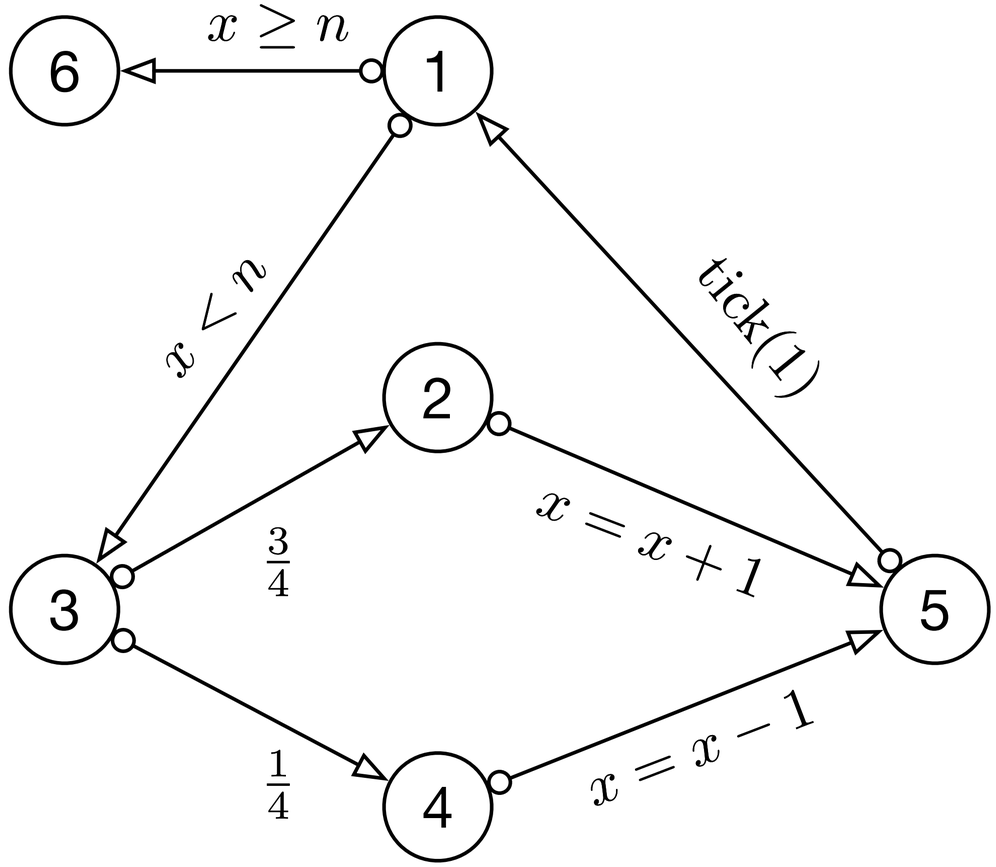
Base Functions

$$M := \mathbf{1} \mid x \mid M_1 \cdot M_2 \mid \max(0, P)$$

Potential & Rewrite Functions

$$P := k \cdot M \mid P_1 + P_2$$

```
1 while (x < n) {
2    x = x + 1
3    [3/4]
4    x = x - 1;
5    tick(1);
6 } //termination point
```



c1 [p] c2 : probabilistic branching

id = e bop R : sampling assignment

R is probability distribution

if * c1 else c2 : non-deterministic choice