Acceptance Sampling and its Use in Probabilistic Verification

Håkan L. S. Younes
Carnegie Mellon University
The Problem

- Let $\phi$ be some property of a system holding with unknown probability $p$
- We want to \textit{approximately} verify the hypothesis $p \geq p'$ using sampling
- This problem comes up in PCTL/CSL model checking: $\Pr_{\geq p'}(\phi)$
  - A sample is the truth value of $\phi$ over a sample execution path of the system
Quantifying “Approximately”

- Probability of accepting the hypothesis $p < p'$ when in fact $p \geq p'$ holds: $\leq \alpha$
- Probability of accepting the hypothesis $p \geq p'$ when in fact $p < p'$ holds: $\leq \beta$
Desired Performance of Test

Actual probability $p' \geq p$ of $\phi$ holding

- Probability of accepting hypothesis $p \geq p'$
- $1 - \alpha$
- $\beta$

False negatives

Unrealistic!

False positives

Actual probability $p$ of $\phi$ holding
Relaxing the Problem

- Use two probability thresholds: $p_0 > p_1$
  - (e.g. specify $p'$ and $\delta$ and set $p_0 = p' + \delta$ and $p_1 = p' - \delta$)

- Probability of accepting the hypothesis $p \leq p_1$ when in fact $p \geq p_0$ holds: $\leq \alpha$

- Probability of accepting the hypothesis $p \geq p_0$ when in fact $p \leq p_1$ holds: $\leq \beta$
Realistic Performance of Test

Actual probability $p$ of $\phi$ holding

Probability of accepting hypothesis $p \geq p_o$

$1 - \alpha$

$\beta$

$p_1$, $p'$, $p_0$

Indifference region

False negatives

False positives

Actual probability $p$ of $\phi$ holding
Method 1: Fixed Number of Samples

- Let $n$ and $c$ be two non-negative integers such that $c < n$
  - Generate $n$ samples
  - Accept the hypothesis $p \leq p_1$ if at most $c$ of the $n$ samples satisfy $\phi$
  - Accept the hypothesis $p \geq p_0$ if more than $c$ of the $n$ samples satisfy $\phi$
Method 1: Choosing $n$ and $c$

- Each sample is a Bernoulli trial with outcome 0 ($\phi$ is false) or 1 ($\phi$ is true)
- The sum of $n$ iid Bernoulli variates has a binomial distribution

\[ F(c, n, p) = \sum_{i=0}^{c} \binom{n}{i} p^i (1 - p)^{n-i} \]
Method 1: Choosing \( n \) and \( c \) (cont.)

- Find \( n \) and \( c \) simultaneously satisfying:
  1. \( \forall p' \in [p_0,1], F(c, n, p_0) \leq \alpha \)
  2. \( \forall p' \in [0,p_1], 1 - F(c, n, p_1) \leq \beta \)

- Non-linear system of inequalities, typically with multiple solutions!
  - Want solution with smallest \( n \)
  - Solve non-linear optimization problem using numerical methods
Method 1: Example

- \( p_0 = 0.5, \ p_1 = 0.3, \ \alpha = 0.2, \ \beta = 0.1: \)
  - Use \( n = 32 \) and \( c = 13 \)
Idea for Improvement

- We can sometimes stop before generating all $n$ samples
  - If after $m$ samples more than $c$ samples satisfy $\phi$, then accept $p \geq p_0$
  - If after $m$ samples only $k$ samples satisfy $\phi$ for $k + (n - m) \leq c$, then accept $p \leq p_1$
- Example of a sequential test
- Can we explore this idea further?
Method 2: Sequential Acceptance Sampling

- Decide after each sample whether to accept $p \geq p_0$ or $p \leq p_1$, or if another sample is needed.
The Sequential Probability Ratio Test [Wald 45]

- An efficient sequential test:
  - After $m$ samples, compute the quantity
    \[
    f = \prod_{i=1}^{m} \frac{\Pr[X = x_i | p = p_1]}{\Pr[X = x_i | p = p_0]}
    \]
  - Accept $p \geq p_0$ if $f \leq \beta/(1 - \alpha)$
  - Accept $p \leq p_1$ if $f \geq (1 - \beta)/\alpha$
  - Otherwise, generate another sample
Method 2: Graphical Representation

- We can find an acceptance line and a rejection line given $p_0, p_1, \alpha, \text{ and } \beta$:

$\phi$  
\[ \text{Number of samples satisfying } \]  
\[ \text{Number of generated samples} \]

- Accept
- Continue sampling
- Reject

$A_{p_0, p_1, \alpha, \beta}(m)$  
$R_{p_0, p_1, \alpha, \beta}(m)$
Method 2: Graphical Representation

- Reject hypothesis $p \geq p_0$ (accept $p \leq p_1$)

![Graphical representation of hypothesis testing](image-url)
Method 2: Graphical Representation

- Accept hypothesis $p \geq p_0$

![Graphical representation of the method](image)
Method 2: Example

- $p_0 = 0.5$, $p_1 = 0.3$, $\alpha = 0.2$, $\beta = 0.1$:
Method 2: Number of Samples

- No upper bound, but terminates with probability one ("almost surely")
- On average requires many fewer samples than a test with fixed number of samples
Method 2: Number of Samples (cont.)

- $p_0 = 0.5$, $p_1 = 0.3$, $\alpha = 0.2$, $\beta = 0.1$:
Acceptance Sampling with Partially Observable Samples

- What if we cannot observe the sample values without error?
  - $\Pr_{\geq 0.5}(\Pr_{\geq 0.7}(\diamond \leq 9 \text{ recharging}) \cup \leq 6 \text{ have tea})$
Acceptance Sampling with Partially Observable Samples

- What if we cannot observe the sample values without error?

True, false, or another sample?
Modeling Observation Error

- Assume prob. $\leq \alpha'$ of observing that $\phi$ does not satisfy a sample when it does.
- Assume prob. $\leq \beta'$ of observing that $\phi$ satisfies a sample when it does not.
Accounting for Observation Error

- Use narrower indifference region:
  - \( p_0' = p_0(1 - \alpha') \)
  - \( p_1' = 1 - (1 - p_1)(1 - \beta') \)
- Works the same for both methods!
**Observation Error: Example**

- $p_0 = 0.5$, $p_1 = 0.3$, $\alpha = 0.2$, $\beta = 0.1$
- $\alpha' = 0.1$, $\beta' = 0.1$

Number of samples satisfying $\phi$

Number of generated samples

Average number of samples

Actual probability $p$ of $\phi$ holding
Application to CSL Model Checking [Younes & Simmons 02]

- Use acceptance sampling to verify probabilistic statements in CSL
- Can handle CSL without steady-state and unbounded until
  - Nested probabilistic operators
  - Negation and conjunction of probabilistic statements
Benefits of Sampling

- Low memory requirements
- Model independent
- Easy to parallelize
- Provides “counter examples”
- Has “anytime” properties
CSL Model Checking Example: Symmetric Polling System

- Single server, \( n \) polling stations
- State space of size \( O(n \cdot 2^n) \)
- Property of interest:
  - When full and serving station 1, probability is at least 0.5 that station 1 is polled within \( t \) time units

```text
Polling stations \( \lambda \) \( \lambda \) \( \lambda \) \( \lambda \) ... \( \lambda \)
Server
```
Symmetric Polling System (results) [Younes et al. ??]

- Verification time (seconds)
- Size of state space

- $T=40$ (symbolic)
- $T=20$ (")
- $T=10$ ("")
- $T=40$ (sampling)
- $T=20$ ("")
- $T=10$ ("")

Pr_{\geq 0.5}(true \leq t_{poll1})

$\alpha=\beta=10^{-2}$
$\delta=10^{-2}$
Symmetric Polling System (results) [Younes et al. ??]

\[
\Pr_{\geq 0.5}(\text{true } U \leq t_{\text{poll1}})
\]

\[
\alpha = \beta = 10^{-2} \\
\delta = 10^{-2}
\]
Symmetric Polling System (results) [Younes et al. ??]

α = β = 10^{-10}

α = β = 10^{-8}

α = β = 10^{-6}

α = β = 10^{-4}

α = β = 10^{-2}

Verification time (seconds)

Pr_{\geq 0.5}(true U_{\leq t} poll1)

n=10

t=50
Notes Regarding Comparison

- Single state vs. all states
- Hypothesis testing vs. probability calculation/estimation
- Bounds on error probability vs. convergence criterion
Relevance to Planning
[Younes et al. 03]

- Planning for CSL goals in continuous-time stochastic domains
- Verification guided policy search:
  - Start with initial policy
  - Verify if policy satisfies goal in initial state
    - Good: return policy as solution
    - Bad: use sample paths to guide policy improvement and iterate
Summary

- Acceptance sampling can be used to verify probabilistic properties of systems
- Have shown method with fixed number of samples and sequential method
- Sequential method better on average and adapts to the difficulty of a problem