Bayes-Optimal Methods for Simulation Optimization

Peter I. Frazier

Operations Research & Information Engineering, Cornell University

Monday April 18, 2011
AFOSR Grantees Meeting
Arlington, VA

Simulation Optimization

Generically stated, simulation optimization is the problem:

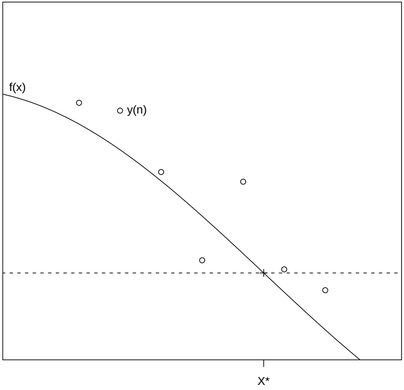
$$\max_x f(x)$$

where

- f cannot be evaluated analytically. It must instead be evaluated through Monte Carlo simulation.
- We also assume the simulation is takes a long time to run, so our goal is to do as well as we can with a limited budget of function evaluations (computationally expensive simulation).

Stochastic Root Finding

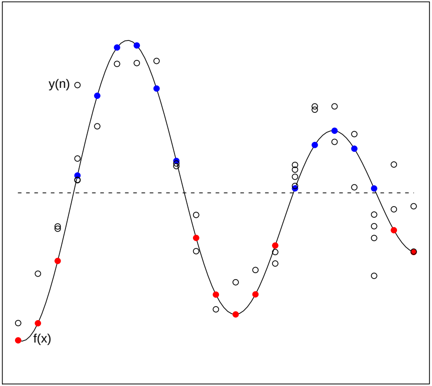
Find the root of a monotone function whose values can only be measured with noise. Use as few function evaluations as possible.



This is equivalent to maximizing a concave function whose gradient values are measured with noise.

Multiple Comparisons with a Standard

Identify those points x whose value $f(x)$ is above a threshold, when function evaluations are noisy. Use as few function evaluations as possible.



In discrete domains, this is **multiple comparisons with a standard**.
In continuous domains, this is **finding level sets**.

Overview

- Simulation optimization problems can be formulated in a Bayesian decision theoretic framework.
- When formulated in this way, optimal policies can be characterized using dynamic programming.
- (This is well known, although perhaps not obvious).
- Novel results:
  - We find the Bayes-optimal policy for an idealized version of stochastic root-finding and apply this to the real stochastic root-finding problem.
  - We find the Bayes-optimal policy for sequential multiple comparisons with a standard.

Outline

General Decision-Theoretic Framework

1. Before we begin, nature fixes some true function f.
   - e.g., f is the function whose root we are trying to find.
2. For time $n = 1$ up to N, we:
   1. Choose some type of measurement x_n to perform.
   2. Observe some value y_n whose distribution depends on x_n and f.
3. After the final measurement, we make some decision $\widehat{x_*}$ based on the data $x_1, \ldots, x_N$, $y_1, \ldots, y_N$.
4. We pay some penalty $L(\widehat{x_*}, f)$.

Our goal is to choose $x_1, \ldots, x_N, \widehat{x_*}$ adaptively to minimize $L(\widehat{x_*}, f)$.

General Decision-Theoretic Framework

- $\mathbb{E}^{\pi}[L(\widehat{x_*}, f) \mid f]$ is the expected loss when we use policy π and the truth is f.
  - A policy π is a method for choosing each x_{n+1} as a function of $x_{1:n}, y_{1:n}$, and choosing $\widehat{x_*}$ as a function of $x_{1:N}, y_{1:N}$.
- We cannot solve $\inf_{\pi} \mathbb{E}^{\pi}[L(\widehat{x_*}, f) \mid f]$ because the answer depends on f, which we do not know.
- In this talk, we are interested in **average case performance**.

General Decision-Theoretic Framework

- The average-case loss under a probability distribution $\mathbb{P}_0$ is

$$\int \mathbb{E}^{\pi}\left[L(\widehat{x}_*, f) \mid f\right] \mathbb{P}_0(df) = \mathbb{E}^{\pi}\left[L(\widehat{x}_*, f)\right]$$

- $\mathbb{P}_0$ is our **prior**, and it may represent our subjective belief about f. It is the measure under which we evaluate average-case performance.
- Goal: minimize the average-case loss

$$\inf_{\pi} \mathbb{E}^{\pi}\left[L(\widehat{x}_*, f)\right]$$

General Decision-Theoretic Framework

- Associated with the prior is a sequence of **posterior** distributions

$$\mathbb{P}_n(f \in \cdot) = \mathbb{P}_0 \left\{ f \in \cdot \mid x_{1:n}, y_{1:n} \right\}.$$

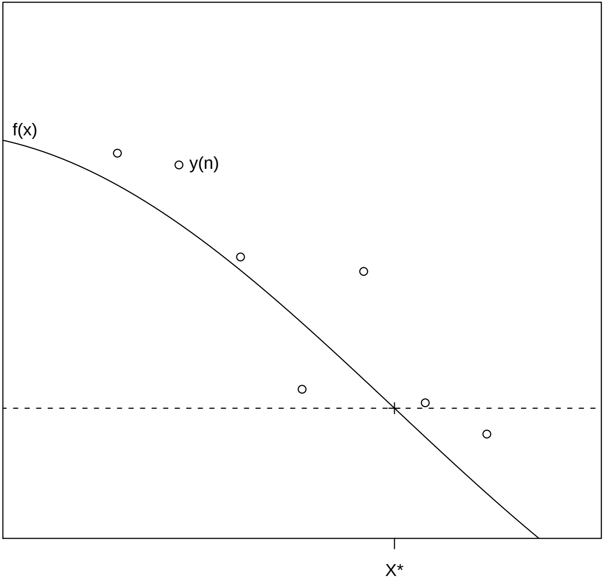
- The problem $\inf_\pi \mathbb{E}^\pi \left[L(\widehat{x}_*, f) \right]$ can be viewed as a **Markov decision process**, where the Markov process being controlled is the sequence of posterior distributions $(\mathbb{P}_n : n \geq 0)$.
- As a Markov decision process, its solution is characterized by the **dynamic programming equations**.

Dynamic Programming

- To solve the DP numerically, we must compute the value function V_n for every possible posterior distribution $\mathbb{P}_n$.
- This is usually **much too large to solve numerically** via brute force.
- Finding analytic solutions to dynamic programs is usually also difficult.
- Approximate DP might be fruitful, but we do not pursue it here.
- In this talk, dynamic programming is used as a theoretical tool.
  - In stochastic root-finding we solve the DP analytically.
  - In multiple comparisons with a standard, we use analytic techniques to decompose the original DP into a collection of easy-to-solve problems.

Outline

Stochastic Root-Finding

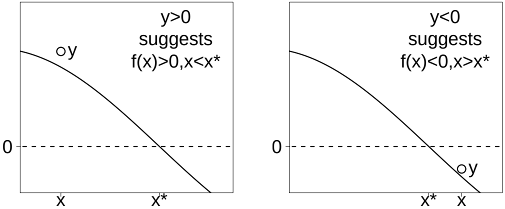


Stochastic Root-Finding

- $f : \mathbb{R} \mapsto \mathbb{R}$ is a decreasing function.
- The only way to evaluate f is via stochastic simulation.
- We observe $y_n = f(x_n) + \varepsilon_n$, where ε_n is independent noise.
- Our goal is to find a root x_*, i.e., a point x_* such that $f(x_*) = 0$.
- Central Question: Given a budget of N measurements, $x_1, \ldots, x_N$, how should we place them to find x_* as accurately as possible?

Stochastic Root-Finding: Model

- Assume we observe only whether y_n is larger or smaller than 0.



- Assume nature gives the incorrect sign with fixed known probability q:

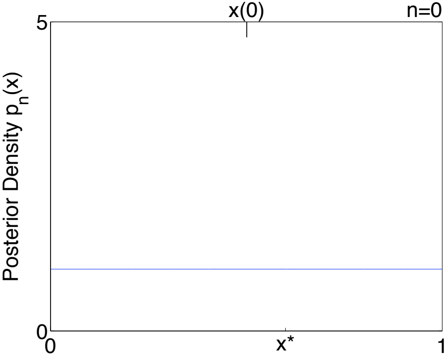
$$\operatorname{sgn}(y_n) = \begin{cases} -\operatorname{sgn}(f(x_n)), & \text{with probability } q, \\ \operatorname{sgn}(f(x_n)), & \text{with probability } 1-q \end{cases}$$

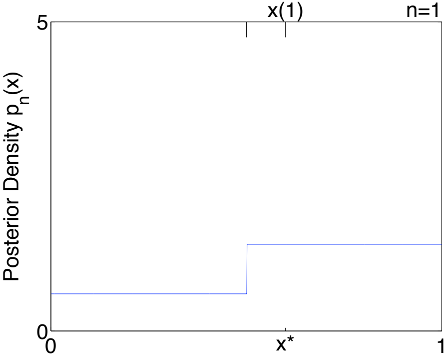
- Ongoing work examines how we can relax these assumptions.

Posterior Distributions

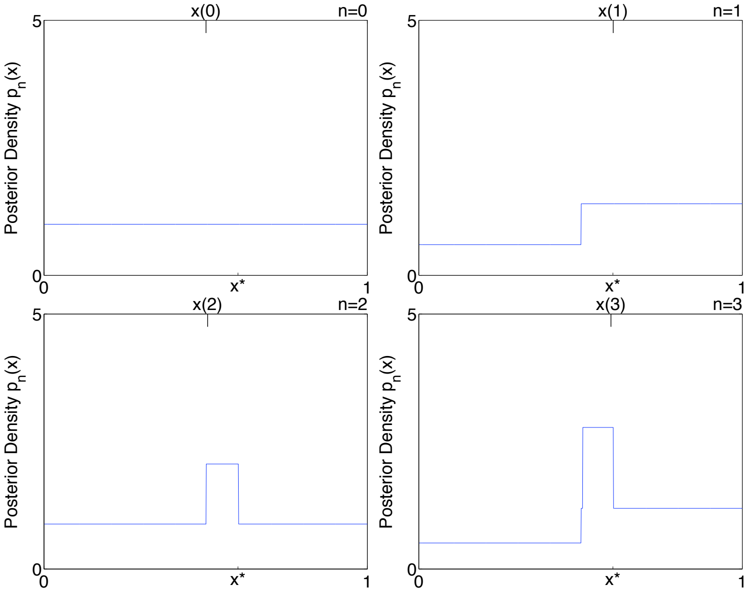
- Place a prior density p_0 on the root x_*, e.g., uniform on $[0,1]$.
- Each measurement x_n produces a new posterior density p_n on x_*:

$$p_n(x) = \mathbb{P}\{X_* \in dx \mid x_{1:n-1}, y_{1:n-1}\}$$





Posterior Distributions



Stochastic Root-Finding

- Our goal is to minimize the final entropy

$$H(p_N) = -\int p_N(x) \log p_N(x)\, dx.$$

- The value function is

$$V_n(p_n) = \inf_{\pi} \mathbb{E}^{\pi} \left[H(p_N) \mid p_n \right],$$

  and the stochastic optimization problem we need to solve is to find the policy π that attains the infimum defining $V_0(p_0)$.
- Bellman's recursion can be written, $V_n(p_n) = \inf_{x_n \in [0,1]} \mathbb{E}\left[V_{n+1}(p_{n+1}) \mid p_n \right]$.

Main Result: Bayes Optimality

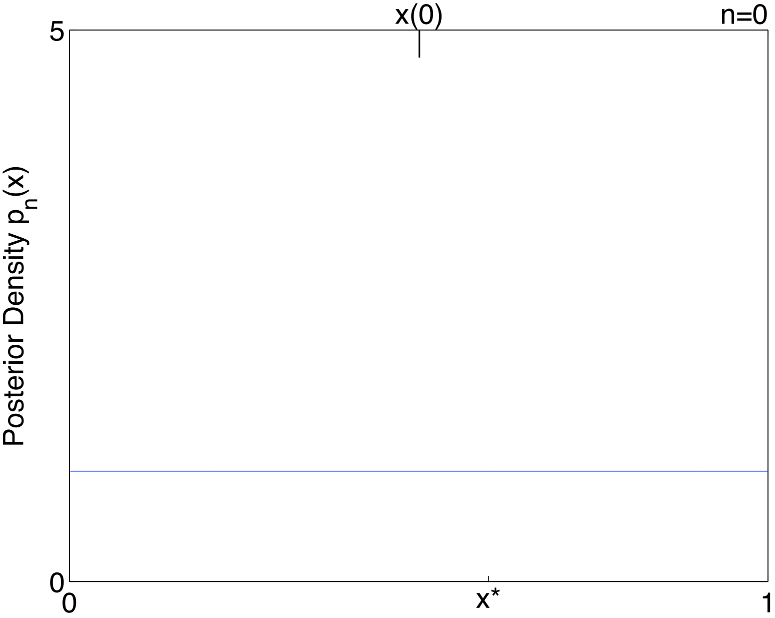
Theorem

The value function can be written explicitly as

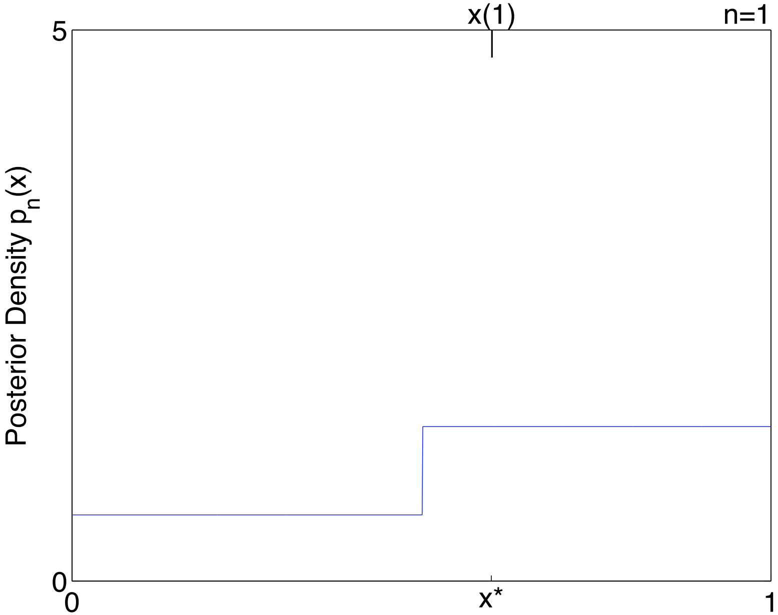
$$V(p_n) = H(p_n) - (N-n)\left[-q\log_2(q) - (1-q)\log_2(1-q)\right],$$

and **the policy that chooses x_n at the median of p_n is optimal**.

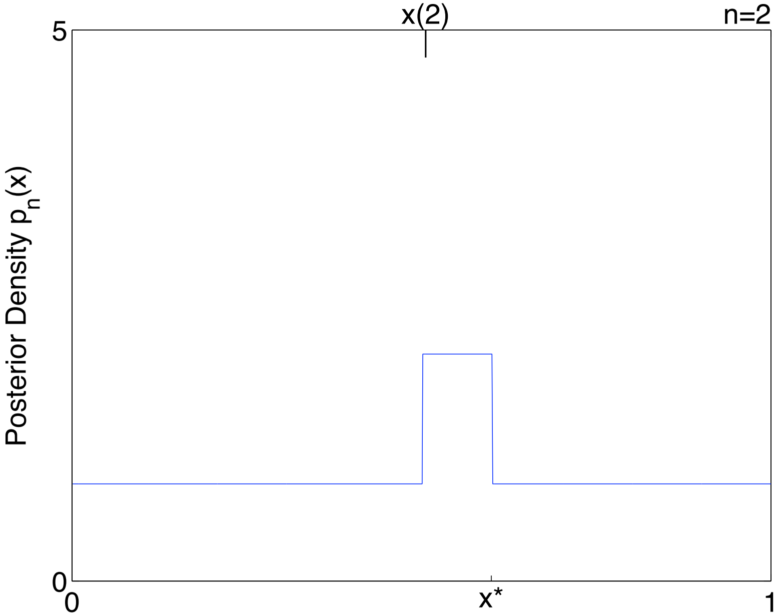
Example



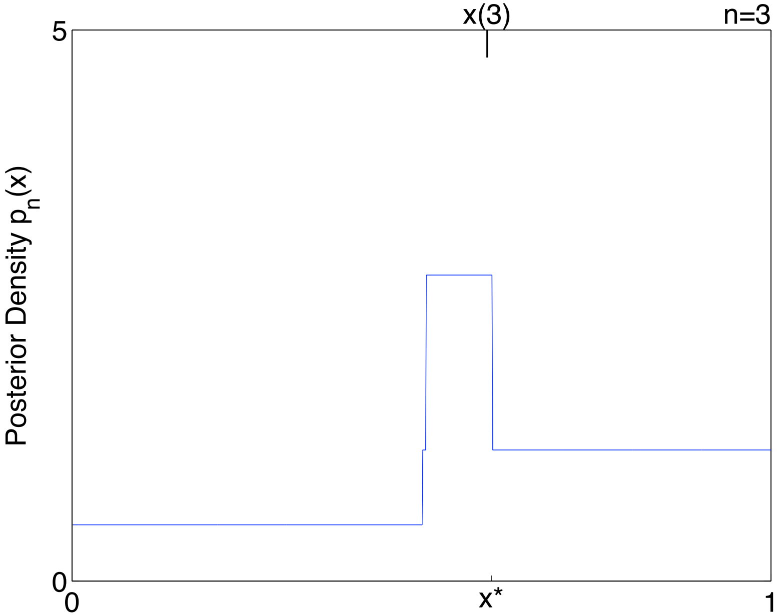
Example



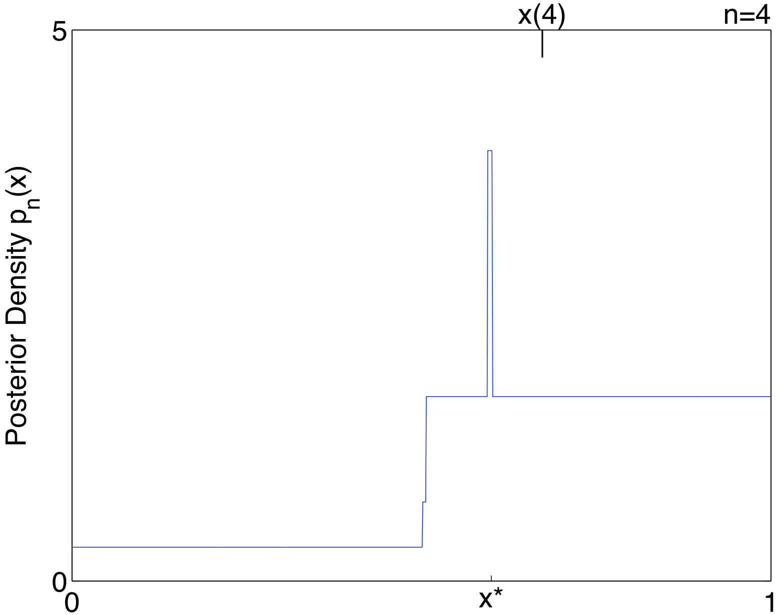
Example



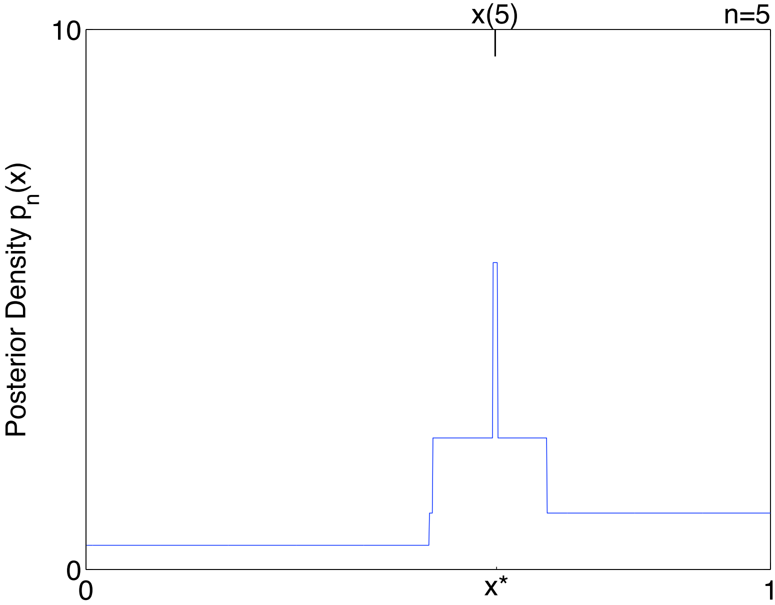
Example



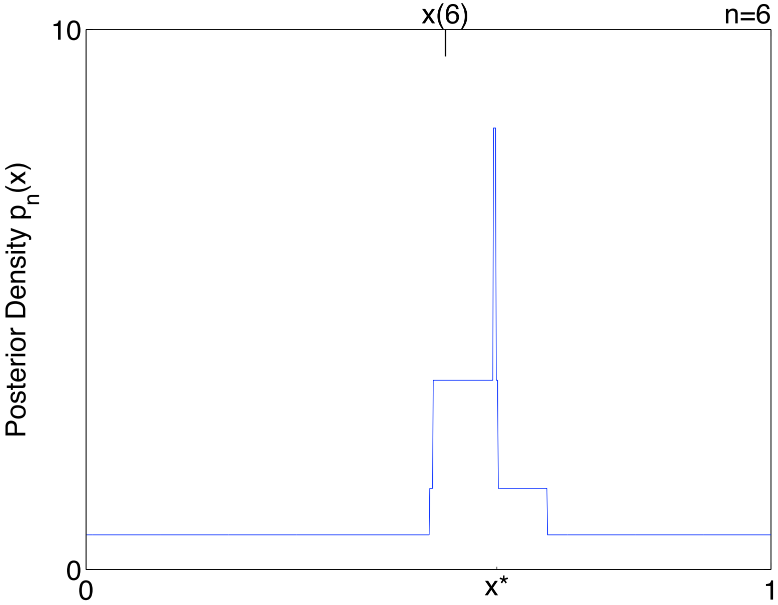
Example



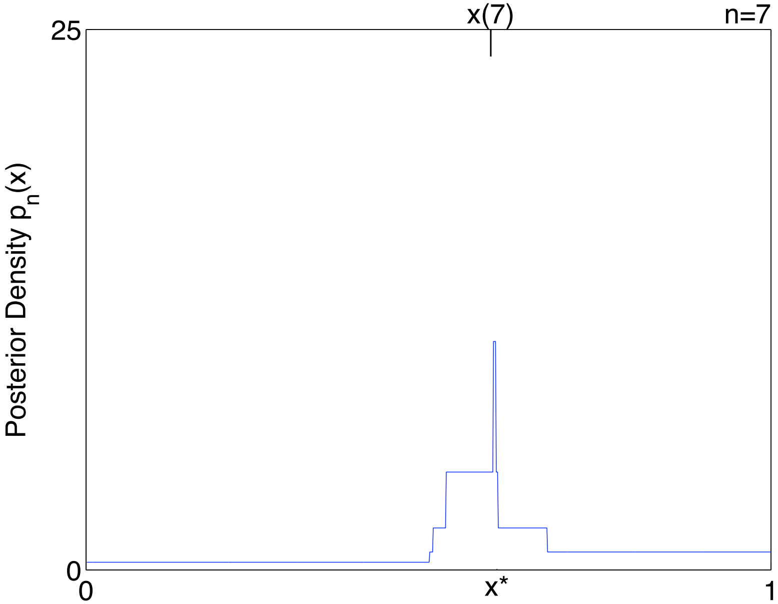
Example



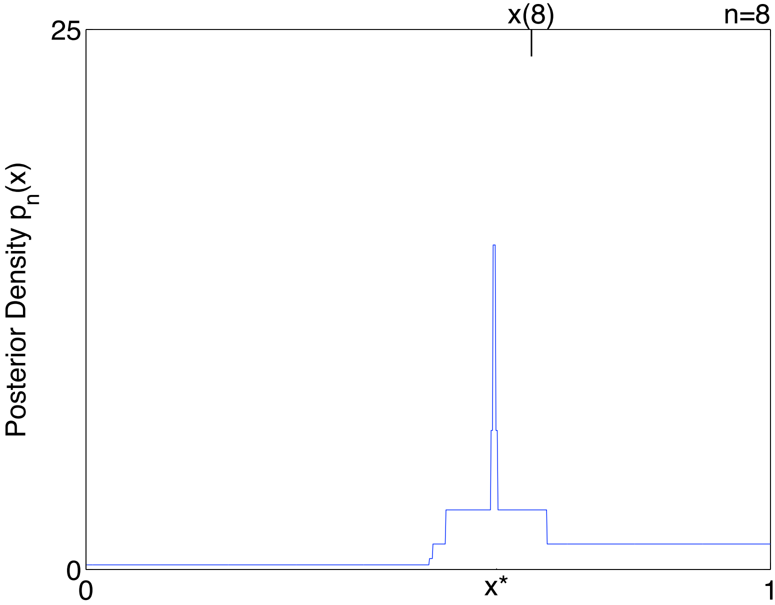
Example



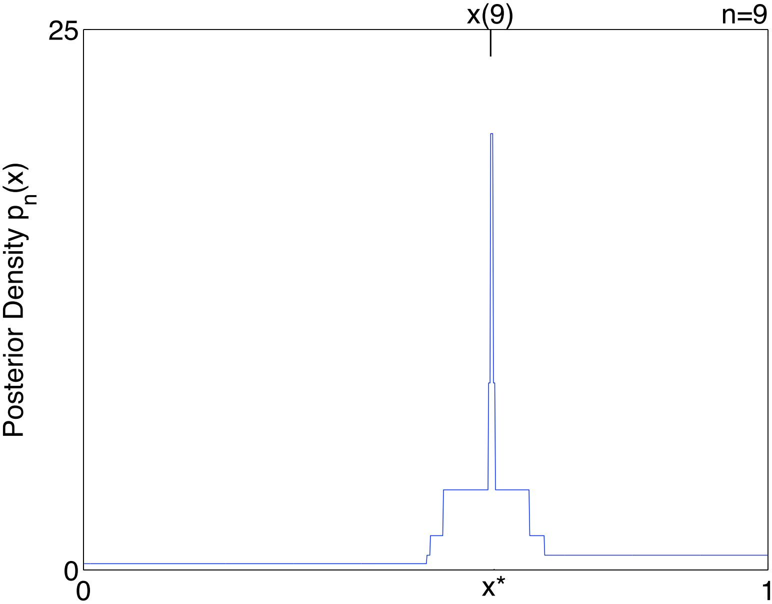
Example



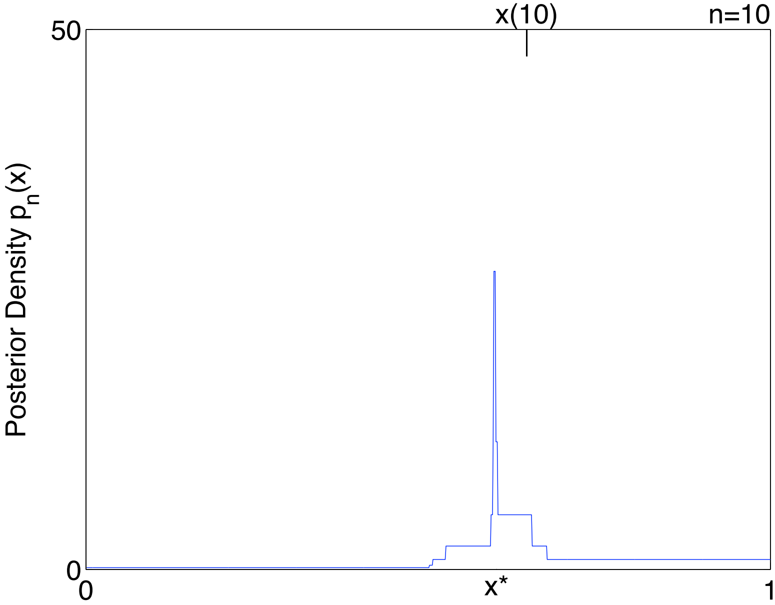
Example



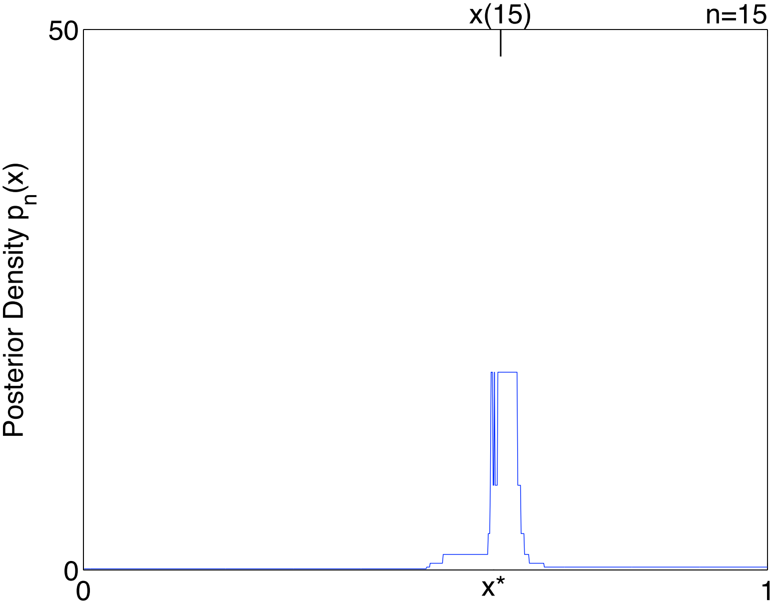
Example



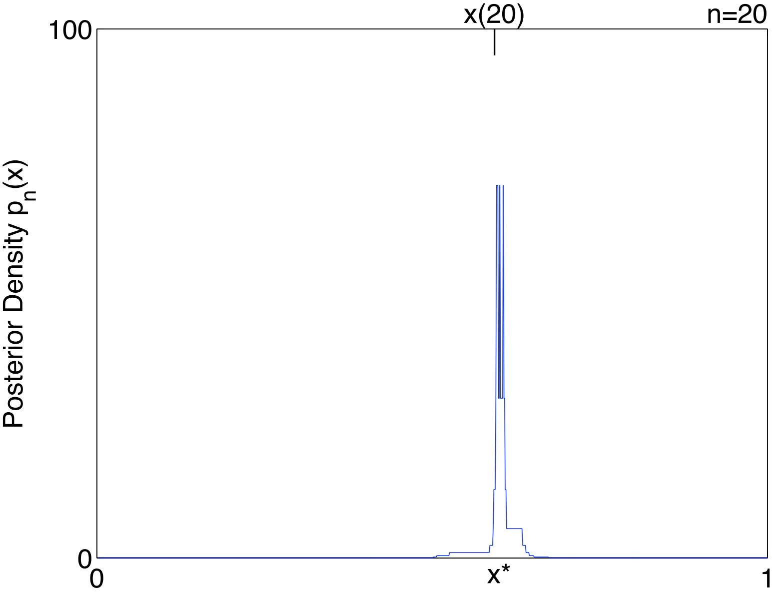
Example



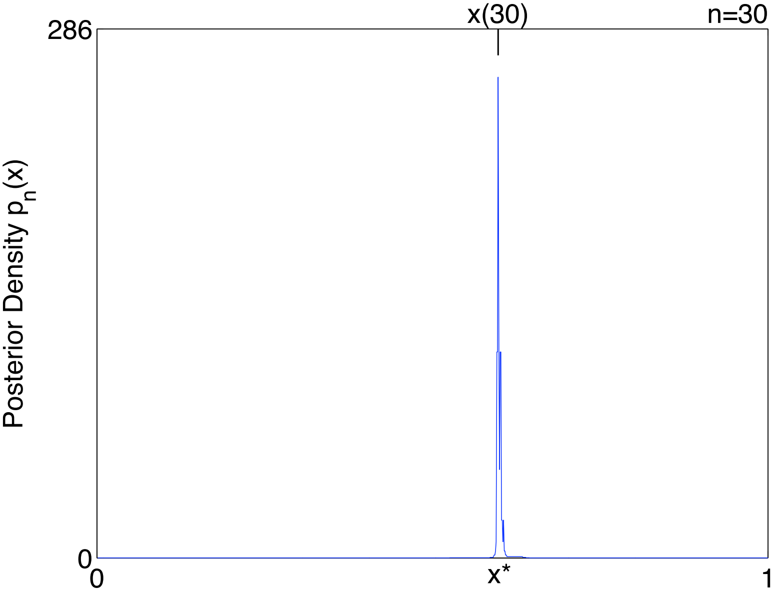
Example



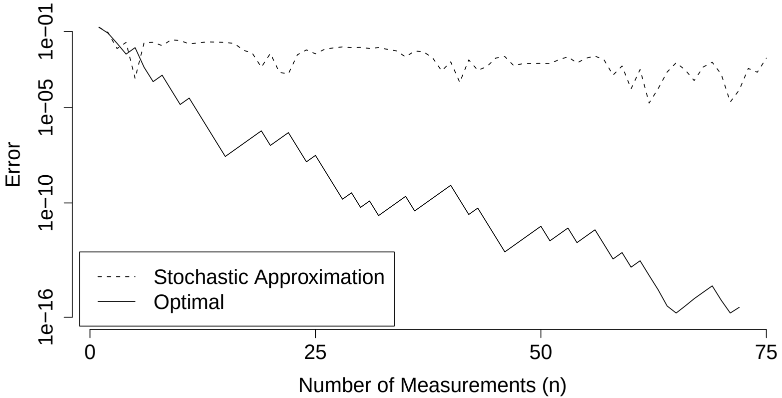
Example



Example



Experimental Results



Performance on a problem with $f(x) = \exp(x) - \exp(1/3)$ and domain $[0,1]$. Error is $|x_n - x_*|$ on one sample path. Stochastic approximation uses stepsize $1/n$.

Geometric Convergence

- q is fixed.
- Updates are done using q.
- Then x_n converges geometrically to the root x_*.

Theorem

Fix $\varepsilon > 0$ *and let* $A_\varepsilon = [x_* - \varepsilon, x_* + \varepsilon]$*.*
Let $\tau = \inf\{n : x_n \in A_\varepsilon\}$*.*
Let $c(q) = 1 + q\log_2(q) + (1-q)\log_2(1-q)$*. Then*

$$\mathbb{E}[\tau] \leq -\log(2\varepsilon)/c(q)$$

[This result is from joint work with Shane Henderson]

Ongoing Work: Unknown Non-Constant $q(x)$

In practice, the error probability $q(x)$ is unknown and varies with x.
To address this, we are investigating several approaches, including:

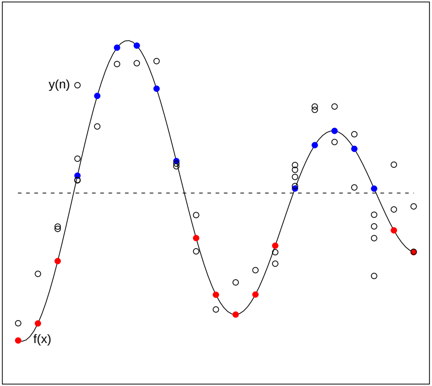
- Fix a value $\bar{q} > 1/2$ and an error tolerance $\varepsilon > 0$.
- At each point x_n, take multiple samples in a sequential fashion to estimate $q(x)$ and to construct a composite bit y_n that is correct with at least probability $\bar{q}$.
- Stop when the estimated $q(x)$ is within ε of $1/2$.

[Joint work with Shane Henderson and Rolf Waeber]

Outline

Multiple Comparisons with a Standard

We now consider a different problem:
Given a limited budget of function evaluations, determine which alternatives have value exceeding a threshold.



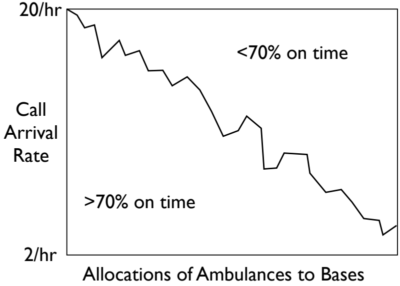
[Joint work with Jing Xie, PhD student at Cornell University]

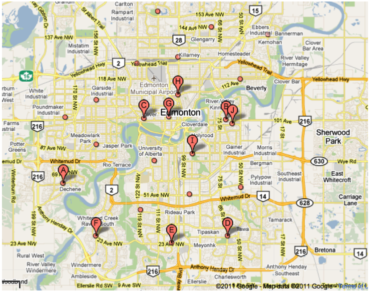
Problem Formulation

- Given alternatives $x = 1, \ldots, k$ and a threshold $d \in \mathbb{R}$.
- Samples from alternative x are $\mathrm{Normal}(\mu_x, \sigma_x^2)$.
- μ_x is unknown, while σ_x^2 is assumed known (can be relaxed).
- Our prior on each μ_x is independent and normal.
- Sampling continues until an external deadline N requires it to stop.
- The algorithm assumes N is unknown and geometrically distributed.
- When sampling stops, we estimate the level set $\{x : \mu_x > d\}$ based on the samples. The reward is the number of alternatives correctly classified.

Example

Administrators are considering allocations of ambulances across 11 bases in the city of Edmonton. They want to know which allocations satisfy mandated minimums for percentage of calls answered in time, under a variety of different possible call arrival rates.





[Thanks to Shane Henderson and Matt Maxwell for providing the ambulance simulation]

Optimal Solution

- The expected reward is the expected number of alternatives correctly classified at the end.
- We decompose this expected reward into an infinite sum of discounted expected one-step rewards

$$R_0 + \mathbb{E}^{\pi}\left[\sum_{n=1}^{\infty} \alpha^{n-1} R_n\right].$$

Here,

- α is the parameter of the geometric distribution governing N.
- R_0 is the expected reward if we stop after taking no samples.
- R_n is the expected one-step improvement, due to sampling, of the probability of correctly classifying alternative x_n.

Optimal Solution

- Written in this way, the problem becomes a **multi-armed bandit**.
- Results from [Gittins & Jones 1974] show that the optimal solution is

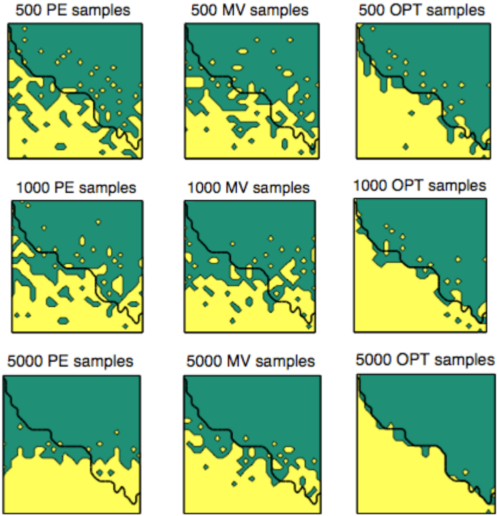
$$\arg\max_x \nu_x(S_{nx}),$$

  where S_{nx} is a parameterization of the marginal posterior on μ_x and $\nu_x(\cdot)$ is the Gittins index.
- The Gittins index $\nu_x(\cdot)$ is defined in terms of a single-alternative version of the problem

$$\nu_x(s) = \sup_{\tau>0} \mathbb{E}\left[\frac{\sum_{n=1}^{\tau} \alpha^{n-1} R_n}{\sum_{n=1}^{\tau} \alpha^{n-1}} \,\middle|\, S_{0x} = s, x_1 = \cdots = x_\tau = x \right].$$

- We can compute Gittins indices efficiently because the single-alternative problem is much smaller than the full DP.

Application to Ambulance Positioning



PE=pure exploration (sample at random);
MV=max variance (equal allocation); OPT=optimal policy.

Conclusion

- **We found the optimal policy** for two types of simulation optimization. Other problems where preliminary results suggest that the DP can be solved:
  - Finding level sets of 1-dimensional continuous functions.
  - Multidimensional stochastic root-finding and noisy convex optimization.
- Having the optimal solutions allows us to **better understand approximate policies**, e.g., knowledge-gradient (KG) policies.
  - The KG policy is optimal for idealized stochastic root-finding.
  - The KG policy is almost optimal for multiple comparisons with normal rewards and no cost of sampling, but performs poorly with Bernoulli rewards or a cost of sampling.
  - This improved understanding will let us use heuristic policies more effectively in problems for which we cannot compute the optimal policy.
- This decision-theoretic approach is a **useful theoretical framework** for developing new and better algorithms for simulation optimization.

Thank You

Any questions?