# Decision Making and Reinforcement under Learning Parameter Uncertainty

Shie Mannor
McGill University

Joint work with: Erick Delage (Stanford), Yaakov Engel (UAlberta), Ron Meir (Technion), Duncan Simester (MIT), Peng Sun (Duke), John Tsitsiklis (MIT)

July 2007

# Agenda

Background and context

Parameter uncertainty: Should we care?

The Bayesian approach

Gaussian processes value estimation

# Context

Machine Learning: Improve performance with more data

Control/Optimization: Find the best solution (policy)

Statistics : Understand the quality of the solution

Data mining: Find structure in data

# Example I: Laptop Power Management

A long-term project with Intel Research

Objective: Save power without annoying the user

Given: Traces of user behavior (120 users $\times$ 3 months $\approx$ 30 years)
Record every 1 second
1B points, each $\approx$100 dimensional

Current state-of-the-art: timeout policies

Validating new policies is not trivial

# Example II: Mail-order Catalog

Catalogs can be shipped every $\approx$2 weeks

Each catalog costs $\approx 1\$$

$\approx 2\text{M}$ customers over 6 years ($\approx 160M$ observations)

Which mailing policy to use?

Objectives:
Short term: Making customers purchase
Long term: Retaining customers

## Decision Making

Classical decision making:

I know where I am

I know what I can do

I know what will happen (or at least the distribution of future events)

# Decision Making

Real-world decision making

I know where I am

I know what I can do

I am not sure what is the distribution of the reward and future events

# Learning = Planning

Planning and learning spectrum

Different knowledge/information models

Small/large state spaces

Simulation/observation

Tractability is key

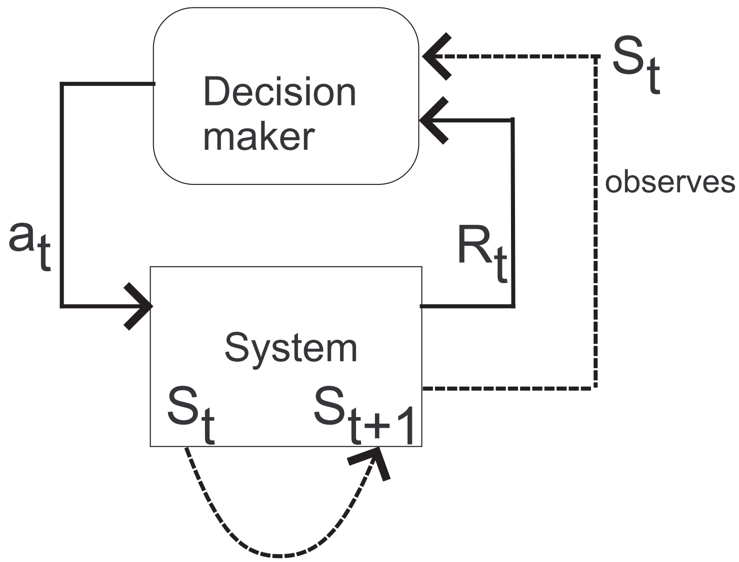
Off/on policy

# Markov Decision Processes

A simple and popular model (MDP)

Ingredients:

1. State space $\mathcal{S}$
2. Action space $\mathcal{A}$
3. Reward $\mathcal{R}$ (a random variable)
4. Transition probability $P(s'|s,a)$.




Dynamics: $S_t \to A_t \to R_t \to S_{t+1}$

# MDPs: The Objective

Objective: maximize (over all policies)

$$\text{Value function} = v(s) = \mathbb{E}^{\pi}\left[\sum_{t=0}^{\infty} \gamma^t R_t \middle| S_0 = s\right]$$

where $\gamma < 1$

There exists an optimal stationary and deterministic policy.

$$\pi : \mathcal{S} \to \mathcal{A}$$

Algorithmically easy: linear programming, policy iteration, value iteration, dynamic programming

# Uncertainties

A single trajectory: inherent uncertainty:
A single customer

Aggregate trajectories: parameter uncertainty:
Average across all customers

Different risk attributes

# Agenda

Background and Context

Parameter Uncertainty: Should we care?

The Bayesian Approach

Gaussian Processes Value Estimation

# Parameter Uncertainty

We always have uncertainty in the parameters

1. I don't have a model - sample from data
2. I know I don't know (part of the model)
3. Things change with time

Probabilistic uncertainty $\Leftrightarrow$ Non-probabilistic uncertainty

# Another Source of Uncertainty

Very high dimensional observation spaces

Examples:
Power management
Mail-order catalog problem

Manageable MDPs are small: $\approx 10,000$ states

Actual MDP represents a simplification - model reduction

## Model Recap

We know: States ($\mathcal{S}$) and actions ($\mathcal{A}$)

But rewards ($\mathcal{R}$) and transitions ($P$) are not known (exactly)

If $\mathcal{S}$ is not known? $\Rrightarrow$ A different talk

Basic question: What are we going to do?

But first - should we care?

# Variance: Illustration

Catalog Circulation Problem

Womens clothing retailer
1.7 million customers $\times$ 4-6 years of mailing/purchase history

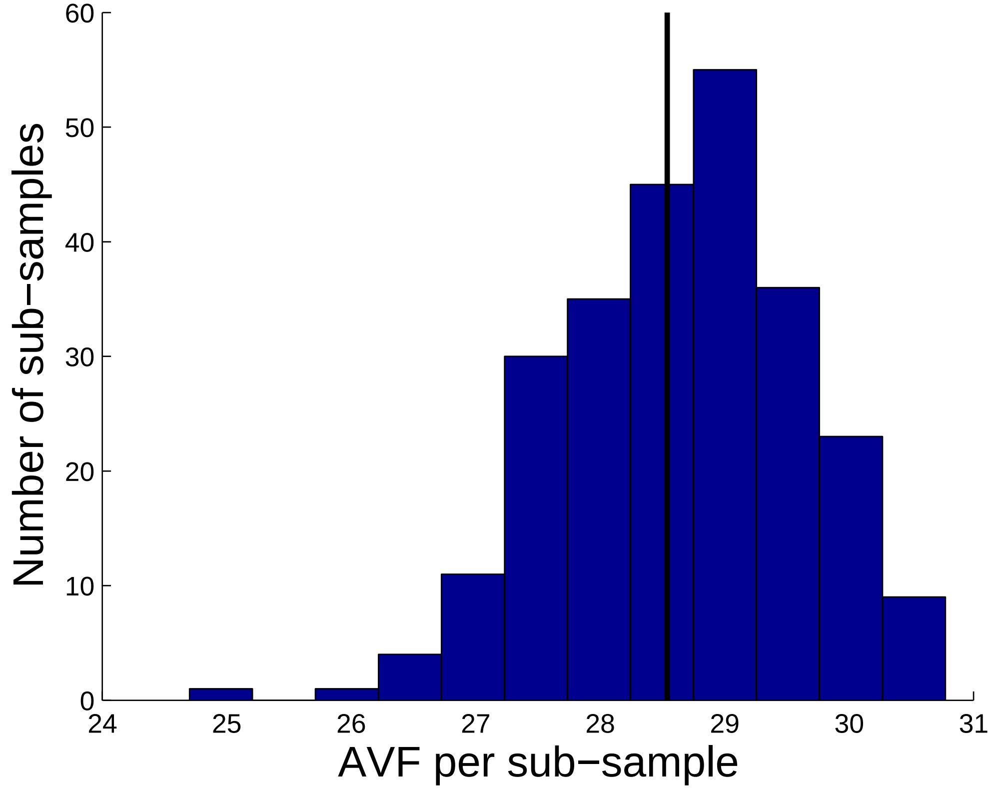
MDP construction: Recency, Frequency and Monetary Value
64 states: Quartiles ($4^3$)

Not a classification problem - need dynamics

250 Sub-samples: 657,000 observations in each

“True” model: All 1.7 million customers

# Value Function: True vs. Estimated




STD = $2

Note: This represents aggregate error

# The Control Problem

Optimization induces additional bias

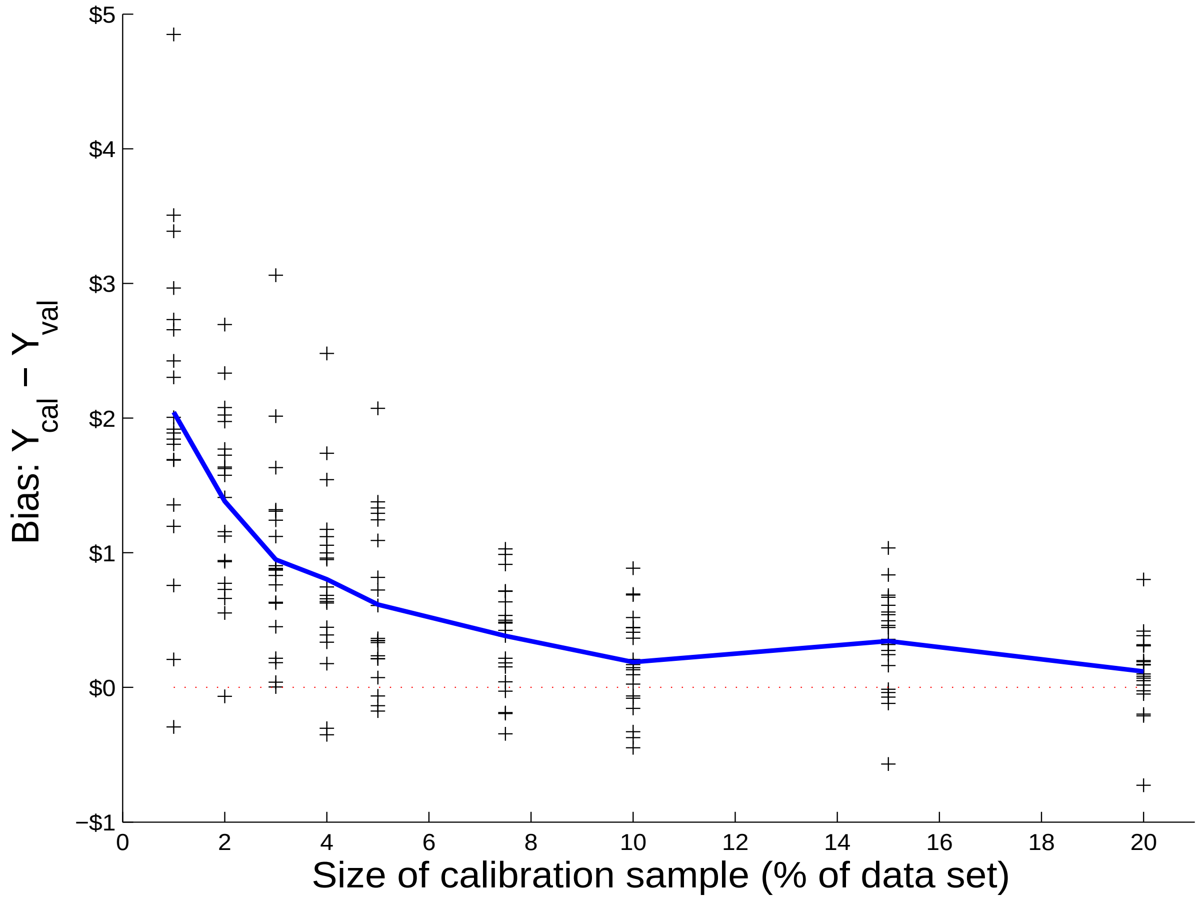
(Jensen's: $X_1, X_2, \ldots, X_n \approx Ber(1/2)$, $\hat{X}_i$ estimates the mean,
$1/2 = \max_i\{\mathbb{E}[\hat{X}_i]\} < \mathbb{E}[\max_i \hat{X}_i]$.)

How big is this bias?

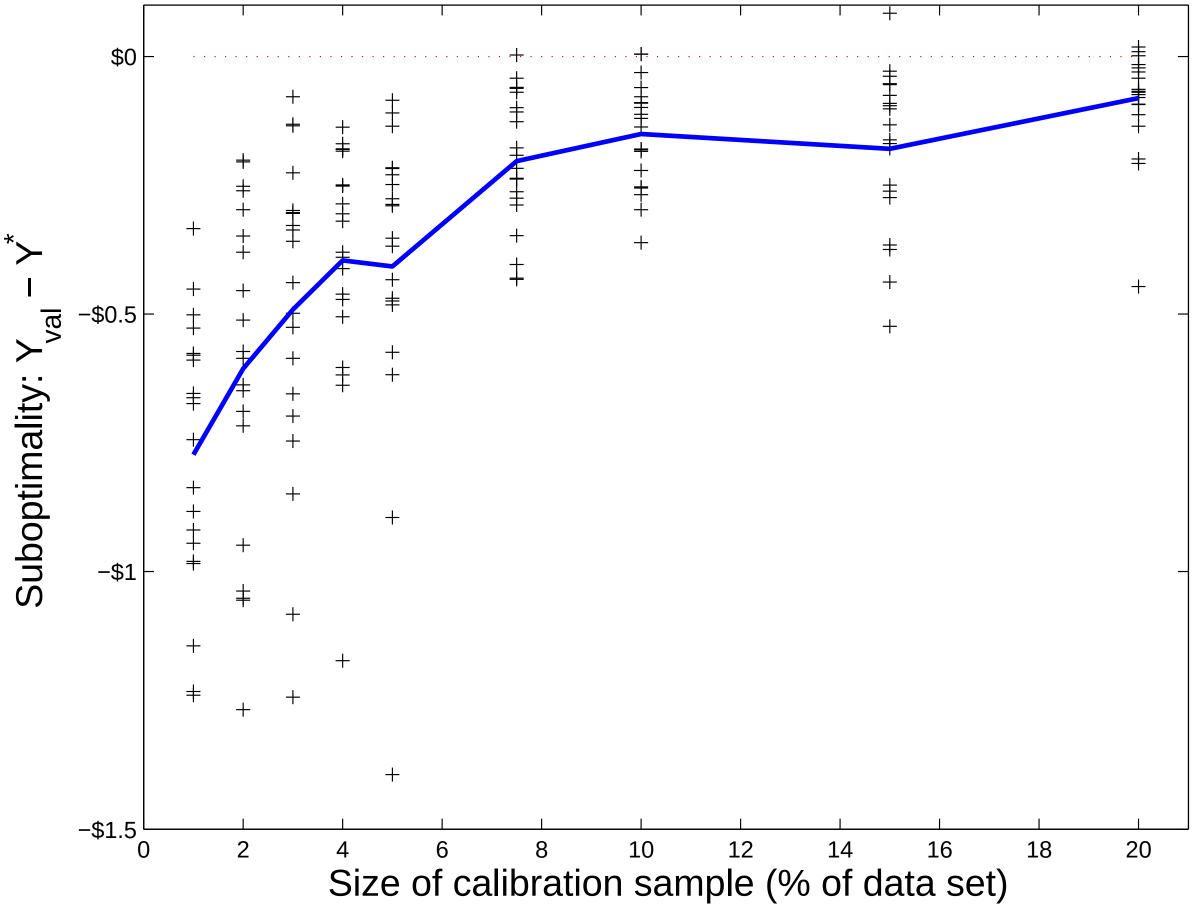
Recipe:

1. Divide data to calibration and validation set
2. Solve on calibration
3. Evaluate on validation
4. Estimate the magnitude of bias

# The Control Problem: Bias



# The Control Problem: Sub-optimality



# Solutions Needed

0. Ignore uncertainty: hope for the best (standard approach in ML/OR)

1. Robustify: expect the worst

2. A Bayesian approach: obtain a probability over models

3. Risk aware approach: optimize performance “most of the time”

# Agenda

Background and context

Parameter uncertainty: Should we care?

The Bayesian approach

Gaussian processes value estimation

# Agenda

Background and context

Parameter uncertainty: Should we care?

The Bayesian approach

Gaussian processes value estimation

# The Bayesian Approach I

Suppose we have a prior on $\mathcal{R}$ and $P$. That is we believe that
$R(s,a) \sim \mathcal{N}$: $P(x;\alpha) = C(\alpha)e^{-(x-\alpha_{mean})^2/\alpha_{var}}$
$P(\cdot|s,a) \sim$ Dirichlet: $\Pr(x|\alpha) = C(\alpha)\Pi_{i=1}^{n} x_i^{\alpha_i - 1}$

After observing data we update our belief

We maintain probability over models

Magic: If we start from $R(s,a) \sim \mathcal{N}$ and $P(\cdot|s,a) \sim$ Dirichlet we maintain the form after the update.

# The Bayesian Approach II

We have a probability over models:

$$\mathbb{E}_{\text{models}} \text{ and } \Pr_{\text{models}}$$

We can now consider $V^\pi(s) = \mathbb{E}^\pi \sum_{t=0}^{\infty} \gamma^t R_t$ as a random variable

For a given $\pi$ and a current belief we can ask what is:

$$\mathbb{E}_{\text{models}}\left[V^\pi(s)\right] = \mathbb{E}_{\text{models}}\left[\mathbb{E}^\pi\left[\sum_{t=0}^{\infty} \gamma^t R_t\right]\right]$$

Mail order catalog: aggregation of customers

# The Bayesian Approach III

We can also ask (percentile optimization):

$$\begin{aligned} \max_{\text{policy } \pi,\, \text{g} \in \mathbb{R}} \quad & g \\ \text{s.t.} \quad & \text{Pr}_{\text{models}} (V^{\pi} e \;\; > g) \geq \rho \end{aligned}$$

Value-at-risk: $\rho$ is the risk parameter.

It turns out that solving the percentile optimization is:

1. NP-hard in general.
2. NP-hard even if transitions are known.

But: For Gaussian reward parameters, problem is polytime.

**Theorem 1** *Percentile optimization is solvable by 2nd order cone programming if there is Gaussian uncertainty in the reward.*
(Delage and Mannor, 2007)

Comparing the computation with "ignoring uncertainty":

Suppose reward $\approx \mathcal{N}(\mu_R, \Theta_R)$ and $q$ is initial distribution on states.

$$\begin{array}{ll} \max\limits_{x \in \mathbb{R}^{|S| \times |A|}} & \sum_a x_a^\top \mu_R - f(\rho) \| \sum_a x_a^\top \Theta_R^{\frac{1}{2}} \|_2 \\ \text{subject to} & \sum_a x_a^\top = q^\top + \sum_a \gamma x_a^\top P_a \\ & x_a^\top \geq 0 \,, \quad \forall \; a \in A. \end{array}$$

Ignoring uncertainty leads to the same problem excluding the red term.

# A Heuristic

Uncertainty in both transitions and rewards

So we can look at the maximization problem.

$$\text{Maximize}_{\text{policy}\,\pi} \mathbb{E}_{\text{models}} \left[ \mathbb{E}^{\pi} \left[ \sum_{t=0}^{\infty} \gamma^t R_t \right] \right]$$
$$\text{equivalent to}:$$
$$\text{Maximize}_{\pi} \mathbb{E}_{\text{models}} \left[ (I - \gamma P_{\pi}^{\text{model}})^{-1} R_{\pi}^{\text{model}} \right]$$

where $P_{\pi}^{\text{model}}$ and $R_{\pi}^{\text{model}}$ are transition probabilities and rewards when using $\pi$ and following the model.

Non-linear expression inside the expectation $\Rrightarrow$ problem is tough.

# Fixed Policy

Can use second order approximation of $(I - \gamma P_\pi^{\text{model}})^{-1}$.

Approximation is good because most third order terms cancel out.

Can obtain (Mannor, Simester, Sun and Tsitsiklis, 2006):
Expressions for the bias and variance estimates

$$\text{Bias} = (I - \gamma \hat{P}_\pi)^{-1} \hat{R}_\pi \; - \mathbb{E}_{\text{models}} \left[ (I - \gamma P_\pi^{\text{model}})^{-1} R_\pi^{\text{model}} \right]$$

Validated on data

Frequentist approach:
Similar bias and variance estimates
CLT like results

# Optimization: More Than a Heuristic

**Theorem 2 (Delage and Mannor 07)** *If one solves:*

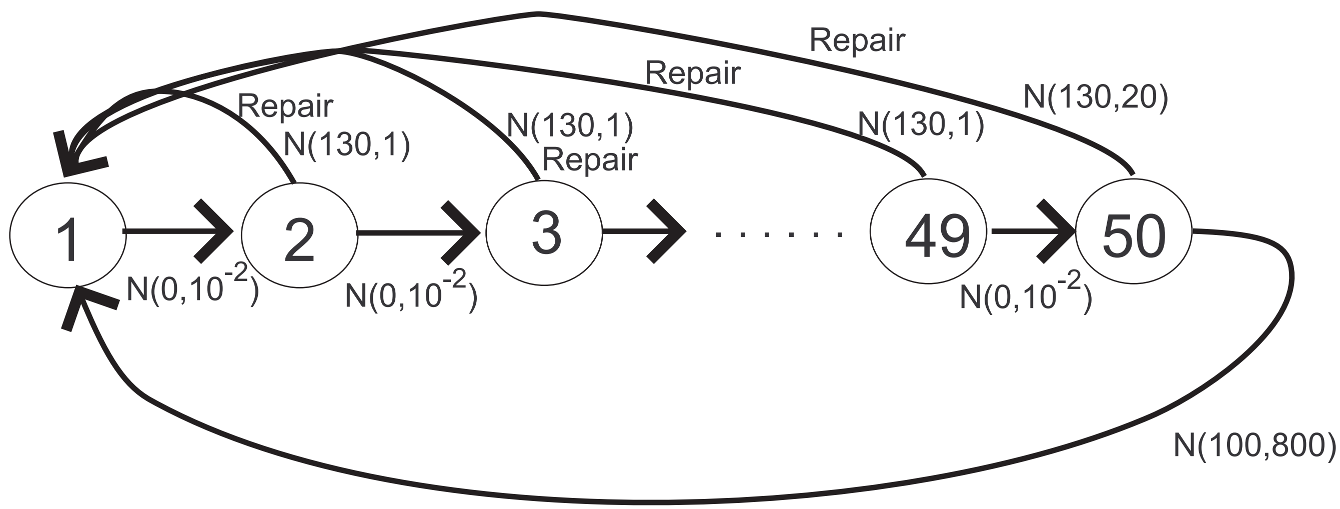
$$\max_{\pi} \mathbb{E}_{\mathsf{models}} \quad [\mathsf{Nominal\,problem} + \mathsf{Second\,order\,terms}]$$

*solution is $o(1/\sqrt{\rho\#_{\mathsf{minimal\,count}}})$ away from the chance-constrained MDP with risk $\rho$.*

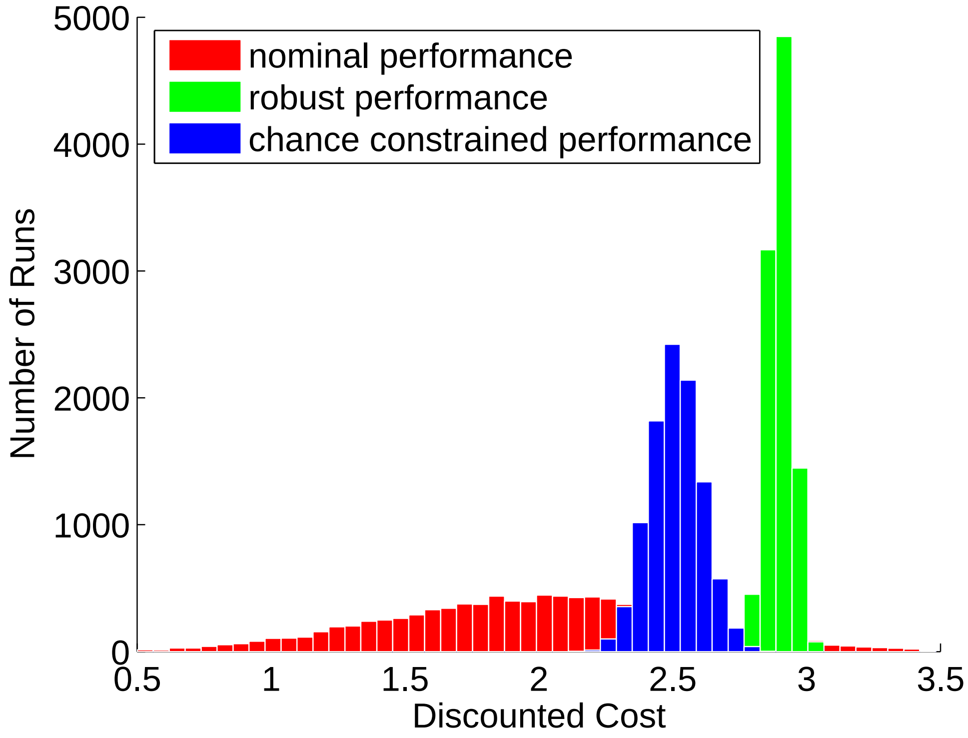
Problem is tractable using modern solvers for $\approx 1,000$ states.

# Results I

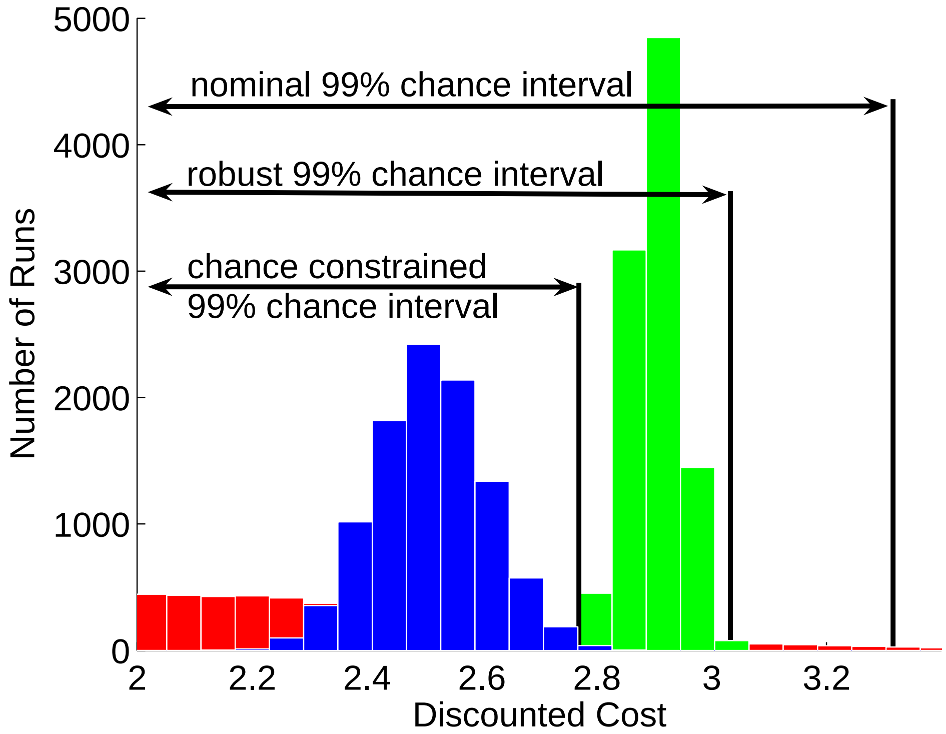
Machine replacement problem (cost minimization)



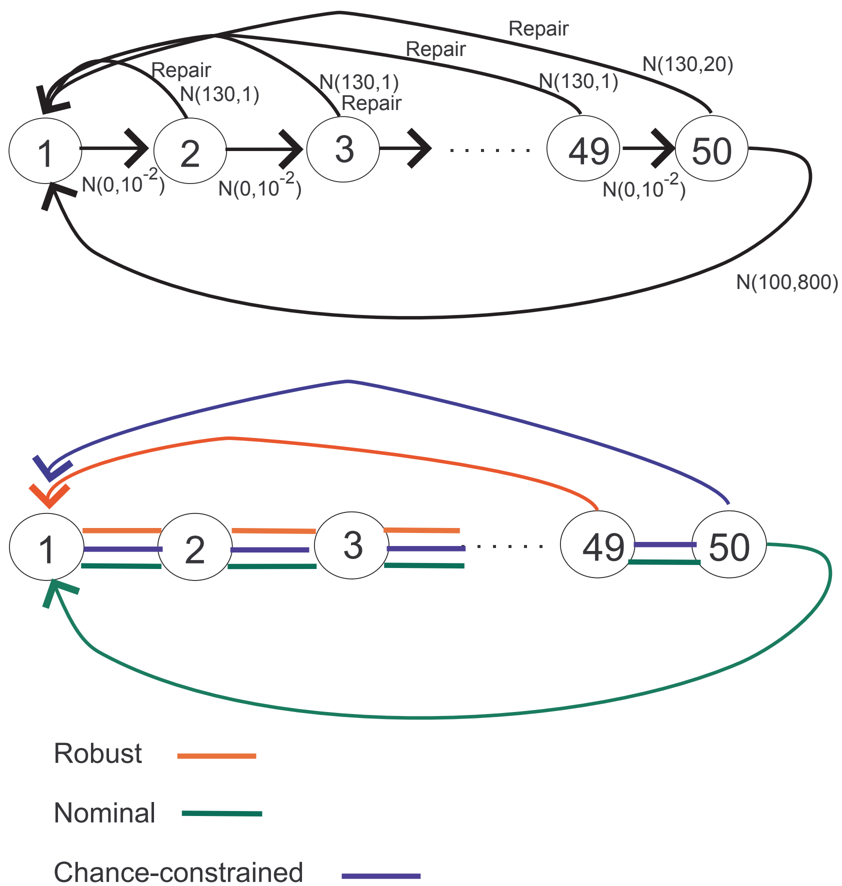
# Results II


nominal performance
robust performance
chance constrained performance
Number of Runs
5000
4000
3000
2000
1000
0
0.5
1
1.5
2
2.5
3
3.5
Discounted Cost

# Results III


nominal 99% chance interval
robust 99% chance interval
chance constrained
99% chance interval
Number of Runs
Discounted Cost
5000
4000
3000
2000
1000
0
2
2.2
2.4
2.6
2.8
3
3.2

# Results IV


Repair
Repair
Repair
N(130,20)
N(130,1)
N(130,1)
Repair
N(130,1)
Repair
1
2
3
49
50
N(0,10^-2)
N(0,10^-2)
N(0,10^-2)
N(100,800)
1
2
3
49
50
Robust
Nominal
Chance-constrained

# Agenda

Background and context

Parameter uncertainty: Should we care?

The Bayesian approach

Gaussian processes value estimation

# Gaussian Processes Value Estimation

But what if we have a big state space?

Consider a fixed policy $\pi$.

The discounted return starting at $s_0 = s$

$$D^\pi(s) = \sum_{i=0}^{\infty} \gamma^i R(s_i, a_i)$$

Therefore:

$$v^\pi(s) = \mathbb{E}_{\text{inherent}}[D^\pi(s)]$$

A simulation problem: We observe rewards and states one by one and want to estimate $v^\pi$.

Monte-Carlo?

Classical approach: look for the value function $v^\pi(s)$:

$$D^\pi(s) = v^\pi(s) + \Delta V^\pi(s)$$

Where to look?

Our approach (Parameter uncertainty): the value is also a random variable:

$$D^\pi(s) = V^\pi(s) + \Delta V^\pi(s)$$

Value function $v^\pi(s) = \mathbb{E}_{\text{models}}\left[V^\pi_{\text{model}}(s)\right]$

By assuming a Gaussian structure on $V^\pi$ we can compute $v^\pi$.

# A Generative Model for the Value

The generative model:

$$\begin{aligned} R(s_t, a_t) &= V(s_t) - \gamma V(s_{t+1}) + N(s_t, s_{t+1}) \\ &= H(s_t, s_{t+1})V + N(s_t, s_{t+1}) \end{aligned}$$

$H$ is a linear integral operator defined by:

$$H(s, s')V = \int d\mathbf{x} \left( \delta(\mathbf{x} - s) - \gamma \delta(\mathbf{x} - s') \right) V(x)$$

Goal:

Find the posterior distribution of $V(\cdot)$, given a sequence of observed states and rewards

# The Prior

Without seeing anything assume $V^\pi(s)$ is a Gaussian process.

Reminder: A Gaussian process is identified by expectation and covariance; its marginal is a Gaussian

$$\begin{aligned} \mathbb{E}_{\text{prior}}[V^\pi(s)] &= 0 \\ \text{Cov}_{\text{prior}}[V^\pi(s, s')] &= \mathbb{E}_{\text{prior}}[V^\pi(s)V^\pi(s')] = k(s, s'), \end{aligned}$$

where $k(s, s')$ is symmetric, positive definite: A Mercer kernel.
(ML blockbuster - support vector machines, kernel regression, etc.)
Indicates prior similarity.

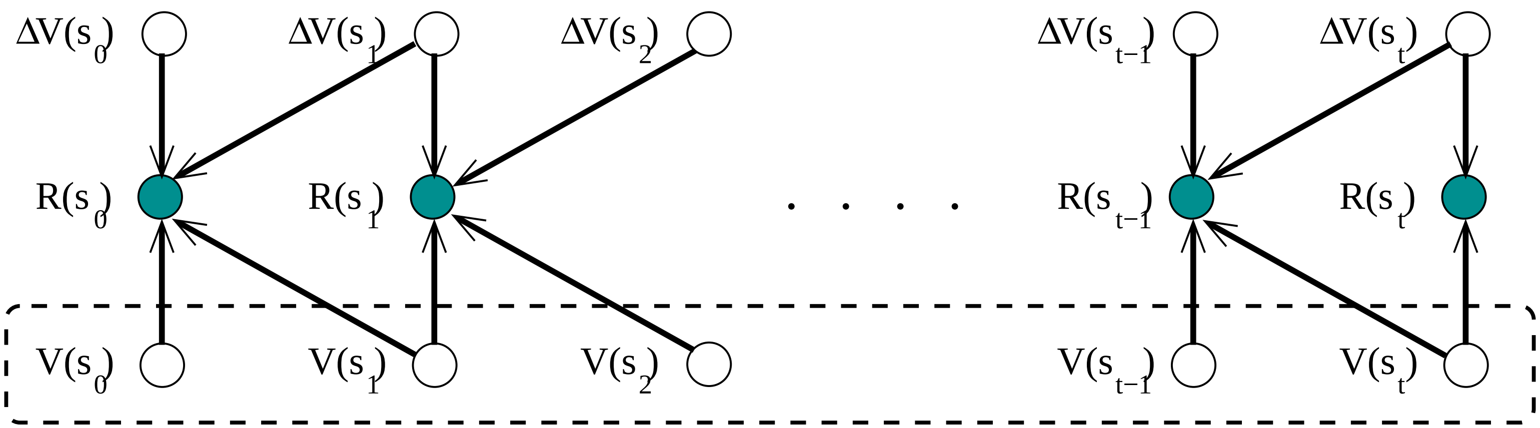
$\Delta V^\pi$ is assumed white IID

Can define the process for any space as long as $k$ is defined.

# Obtaining a Posterior I

With some algebra:

$$R(s_t) = V(s_t) - \gamma V(s_{t+1}) + N(s_t, s_{t+1})$$
$$N(s_t, s_{t+1}) \triangleq \Delta V(s_t) - \gamma \Delta V(s_{t+1})$$



# Obtaining a Posterior II

Problem becomes:

$$R_{t-1} = \mathbf{H}_t V_t + N_t$$

where $R_t = (R(s_0), \ldots, R(s_t))^\top$, $N_t = (N(s_0), \ldots, N(s_{t-1}))^\top$, $V_t = (V(s_0), \ldots, V(s_t))^\top$, and

$$\mathbf{H}_t = \begin{bmatrix} 1 & -\gamma & 0 & \ldots & 0 \\ 0 & 1 & -\gamma & \ldots & 0 \\ \vdots & & & & \vdots \\ 0 & 0 & \ldots & 1 & -\gamma \end{bmatrix}.$$

$N_t$ is colored $\Rightarrow$ a non-standard latent variable computation

# Obtaining a Posterior III

After observing $T$ samples (for every $s$):

$$v^{\pi}(s) = \mathbb{E}_{\text{posterior}}[V^{\pi}(s)] = \sum_{t=1}^{T} k(s_t, s)\alpha_t$$

(expressions for covariance available too).

If kernel behaves well, can truncate sum using a dictionary

$$v^{\pi}(s) = \mathbb{E}_{\text{posterior}}[V^{\pi}(s)] = \sum_{m=1}^{\text{dictionary size}} k(s_m, s)\alpha_m$$

Efficient (temporal-difference) recursive algorithm

# Some Theory

Consistency result: Can get to the true value function with enough data. “A grain of truth theorem”

Can be easily used for exploration

Policy improvement: rollout, slow policy improvement, policy gradients (theory lacking)

Learning is not based on decreasing learning rates

Frequentist: Can re-derive as a least squares solution

# Wrap-up

Parameter uncertainty is a big deal in real-world problems

Small models: Can consider distribution over models

Large models: Can use Gaussian processes to model value process

Does it really work?