Efficient Sampling for Simulator-Defined MDPs

Majid Alkaee Taleghan

Kim Hall

H. Jo Albers (U. Wyoming)

Thomas G. Dietterich

Oregon State University

Supported by NSF grants 0832804 (Computational Sustainability) and 1331932 (CyberSEES)

Outline

- Part 1:
  - Motivating application: Invasive Species in a River Network
  - Brute force solution and examples of the results
- Part 2:
  - Minimizing simulator calls
    - Policy Evaluation
    - Policy Optimization

Invasive Species Management in River Networks

- Tamarisk: invasive tree from the Middle East
  - Has invaded over 3 million acres in the western United States
  - Out-competes native vegetation for water
  - Reduces biodiversity
- What is the best way to manage a spatially-spreading organism?

C.C. Shock, Oregon State University

Existing Approaches in Natural Resource Economics

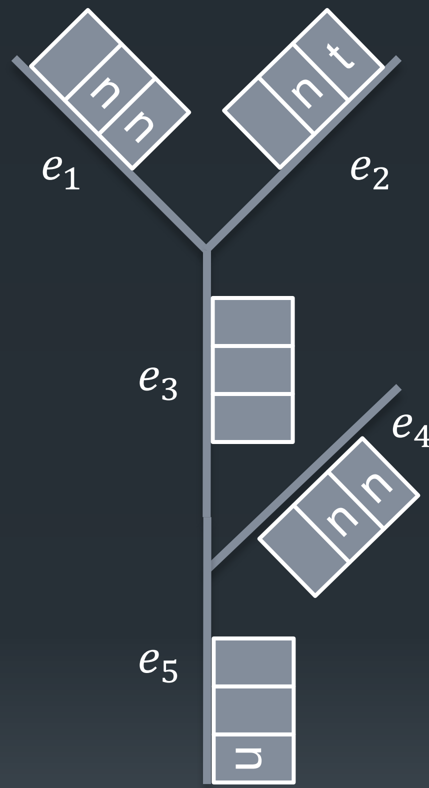
- Model one-dimensional “landscape”
- Spread is only to nearest neighbors
- State variables only consider the presence/absence of the invading species
  - Ignore competition between native and invader
  - Ignore “propagule pressure” (relative abundance and germination success of seeds from different species)
- Resulting optimal policies construct “barriers” to contain the spread

- Some work on more realistic models, but only by replacing stochastic transitions with expectations and treating the system as deterministic.

- Opportunity to advance the field by providing better MDP tools!

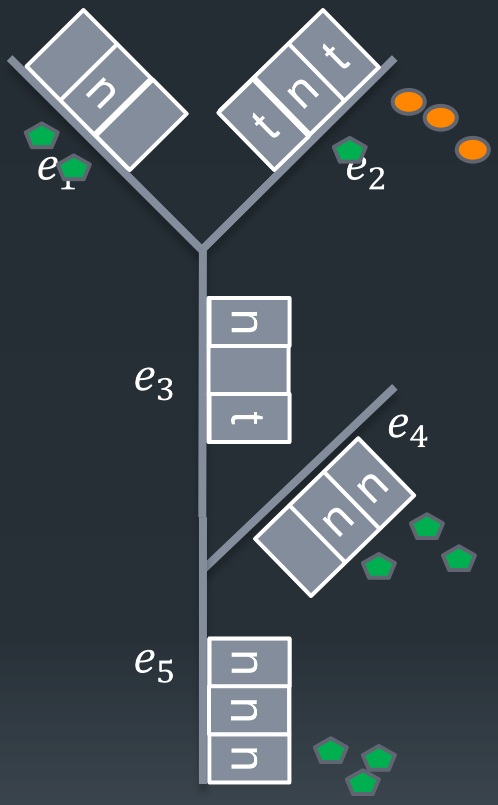
Markov Decision Process

- Tree-structured river network
  - Each edge $e \in E$ has H "sites" where a tree can grow.
  - Each site can be
    - {empty, occupied by native, occupied by invasive}
  - # of states is 3^{EH}
- Management actions
  - Each edge: {do nothing, eradicate, plant, restore (=eradicate + plant)}
  - # of actions is 4^{E}



Dynamics and Objective

- Dynamics:
  - In each time period
    - Natural death
    - Seed production
    - Seed dispersal (preferentially downstream)
    - Seed competition to become established
  - Couples all edges because of spatial spread
  - Inference is intractable



- Objective:
  - Minimize expected discounted costs (cost of invasion + cost of management)
  - Subject to annual budget constraint

Computational Approach

- Transition function can be represented as DBN
- Exact inference in intractable (because we must consider competition from all seeds that arrive at a given slot)
- Sampling is easy

- For each (s, a), draw enough samples to estimate $P(s'|s, a)$ with sufficient accuracy
- Then apply value iteration to solve the MDP

Examples of the Results

- Optimal policy in an edge depends on the state of other edges
  - Case 1: Optimal action is to ERADICATE and then PLANT at the “middle” level
  - Case 2: Optimal action is to ERADICATE and then PLANT in the top left
- Reason?
  - In Case 2, we already have a partial barrier, so there is budget available to plant natives in the top level to protect against eradication failure

Case 1

Case 2

Example of Results (2)

- Exogenous arrivals change the policy
  - seeds of the invader arrive uniformly at random across the landscape (e.g., dropped by birds, transported by fishermen)
- With no exogenous arrivals, if the starting state has an invaded edge, then the optimal policy just performs ERADICATE
- If there are exogenous arrivals, it performs RESTORE.
- In general, under exogenous arrivals, the optimal policy works harder to fill the landscape with native species as a preventative measure

Example of Results (3)

- Prevention is Cheaper than Recovery
  - In an empty river system with exogenous arrivals, the optimal policy PLANTs native species starting upstream and working downstream (if necessary)
  - This is much cheaper than waiting until an invasion arrives and then fighting it via ERADICATION
  - Why: Budget constraints make it impossible to ERADICATE everywhere at once, which allows the invader to spread quickly. Then it can only be slowly eliminated by repeated ERADICATE actions

Summary

- MDP tools can have a big impact in helping ecosystem managers discover and analyze optimal management policies
- Simulator-defined MDPs are a natural way to deal with intractable transition models

Outline

- Part 1:
  - Motivating application: Invasive Species in a River Network
  - Brute force solution and examples of the results
- Part 2:
  - Minimizing simulator calls
    - Policy Evaluation
    - Policy Optimization

More Challenging Setting

- Extremely expensive simulators from ecosystem management problems
- Drawing one sample from these simulators can take more time than performing value iteration on the whole MDP(!)
- We want to minimize the number of calls to the simulator
- We want PAC bounds on the optimality of the policy

Policy Evaluation

- Given:
  - An MDP $\langle S, A, P, R, \gamma \rangle$;
    - $R(s,a) \in [0, R_{max}]; \gamma \in (0,1)$
  - A starting state s_0
  - A fixed policy π
  - A simulator $F: S \times A \mapsto R \times S$ that samples as
    - $R(s,a)$; deterministic
    - $s' \sim P(s'|s,a)$
  - A sampling budget B
- Find:
  - A tight confidence interval on $V^{\pi}(s_0)$
- Notation:
  - $\Delta V^{\pi}(s_0) = V^{\pi}_{upper}(s_0) - V^{\pi}_{lower}(s_0)$ is the width of the confidence interval

Confidence Interval Methods

- Global (full-trajectory) Methods
  - Hoeffding Bound: GCV(H)
  - Empirical Bernstein Bound: GCV(B)
- Local (extended value iteration) Methods
  - Hoeffding Bound: LCVI(H)
  - EBB:LCVI(B)
  - Weissman Multinomial Confidence Region: LCVI(W)

Confidence Interval Methods

- Global methods
  - Choose a depth H
  - Draw $N = \lfloor B/H \rfloor$ trajectories. Let v_i be cumulative discounted return from trajectory i
  - $\hat{V}(s_0) = \frac{1}{N}\sum_{i=1}^{N} v_i$ be the average of these values
  - Compute the confidence interval from $\{v_1, \dots, v_N\}$

Global Hoeffding Bound
(Hoeffding, 1963)

$$V_{upper}(s_0) = \hat{V}(s_0) + V_{max}\sqrt{\frac{\log 2/\delta}{2N}} + \gamma^H V_{max}$$

$$V_{lower}(s_0) = \hat{V}(s_0) - V_{max}\sqrt{\frac{\log 2/\delta}{2N}}$$

$\gamma^H V_{max}$ is the maximum possible reward we lose by truncating the trajectory at depth H

Global Empirical Bernstein Bound (Audibert, Munos, Szepesvari, 2009)

$$V_{upper}(s_0) = \hat{V}(s_0) + \sqrt{\frac{2\widehat{Var}(s_0)\log 3/\delta}{N}} + \frac{3V_{max}\log 3/\delta}{N} + \gamma^H V_{max}$$

$$V_{lower}(s_0) = \hat{V}(s_0) - \sqrt{\frac{2\widehat{Var}(s_0)\log 3/\delta}{N}} - \frac{3V_{max}\log 3/\delta}{N}$$

Here

$$\widehat{Var}(s_0) = \frac{1}{\mathrm{N}}\sum_{i=1}^{N}\left(v_i - \hat{V}(s_0)\right)^2$$

Key idea is that if the variance is small, this can be tighter

Extended Value Iteration with the Local Hoeffding Bound

(Even-Dar, Mannor, Mansour 2003,2006)

At each state s

$$V_{upper}(s) = R(s) + \gamma \sum_{s'} \hat{P}(s'|s) V_{upper}(s') + \gamma V_{max} \sqrt{\frac{\log 2|S|/\delta}{2N(s)}}$$

$$V_{lower}(s) = R(s) + \gamma \sum_{s'} \hat{P}(s'|s) V_{lower}(s') - \gamma V_{max} \sqrt{\frac{\log 2|S|/\delta}{2N(s)}}$$

Perform value iteration on these formulas. The bounds on s_0 give the desired confidence interval

Extended VI with EBB

At each state s

$$V_{upper}(s) = R(s) + \gamma \sum_{s'} \hat{P}(s'|s) V_{upper}(s') + \sqrt{\frac{2\widehat{Var}_{upper}(s) \log 3|S|/\delta}{N(s)}} + \frac{3\gamma V_{max} \log 3|S|/\delta}{N(s)}$$

$$V_{lower}(s) = R(s) + \gamma \sum_{s'} \hat{P}(s'|s) V_{lower}(s') - \sqrt{\frac{2\widehat{Var}_{lower}(s) \log 3|S|/\delta}{N(s)}} - \frac{3\gamma V_{max} \log 3|S|/\delta}{N(s)}$$

Perform value iteration on these formulas. The bounds on s_0 give the desired confidence interval

Weissman L1 Confidence Interval on the Multinomial Distribution
(Weissman et al., 2003)

Given the counts $N(s, s')$ for state s, compute

$$\hat{P}(s'|s) = \frac{N(s,s')}{N(s)}$$

Define a confidence interval

$$CI(N, \delta) = \{\tilde{P} \,|\, \|\tilde{P}(\cdot\,|s) - \hat{P}(\cdot\,|s)\|_1 < \omega\}$$

where

$$\omega = \sqrt{\frac{2[\log(2^{|S|} - 2) - \log \delta / |S|]}{N(s)}}$$

Extended VI with Weissman Multinomial Confidence Interval

(Strehl & Littman, 2004; 2008)

$$V_{upper}(s) = R(s) + \gamma \max_{\tilde{P} \in CI} \sum_{s'} \tilde{P}(s'|s) V_{upper}(s')$$

$$V_{lower}(s) = R(s) + \gamma \min_{\tilde{P} \in CI} \sum_{s'} \tilde{P}(s'|s) V_{lower}(s')$$

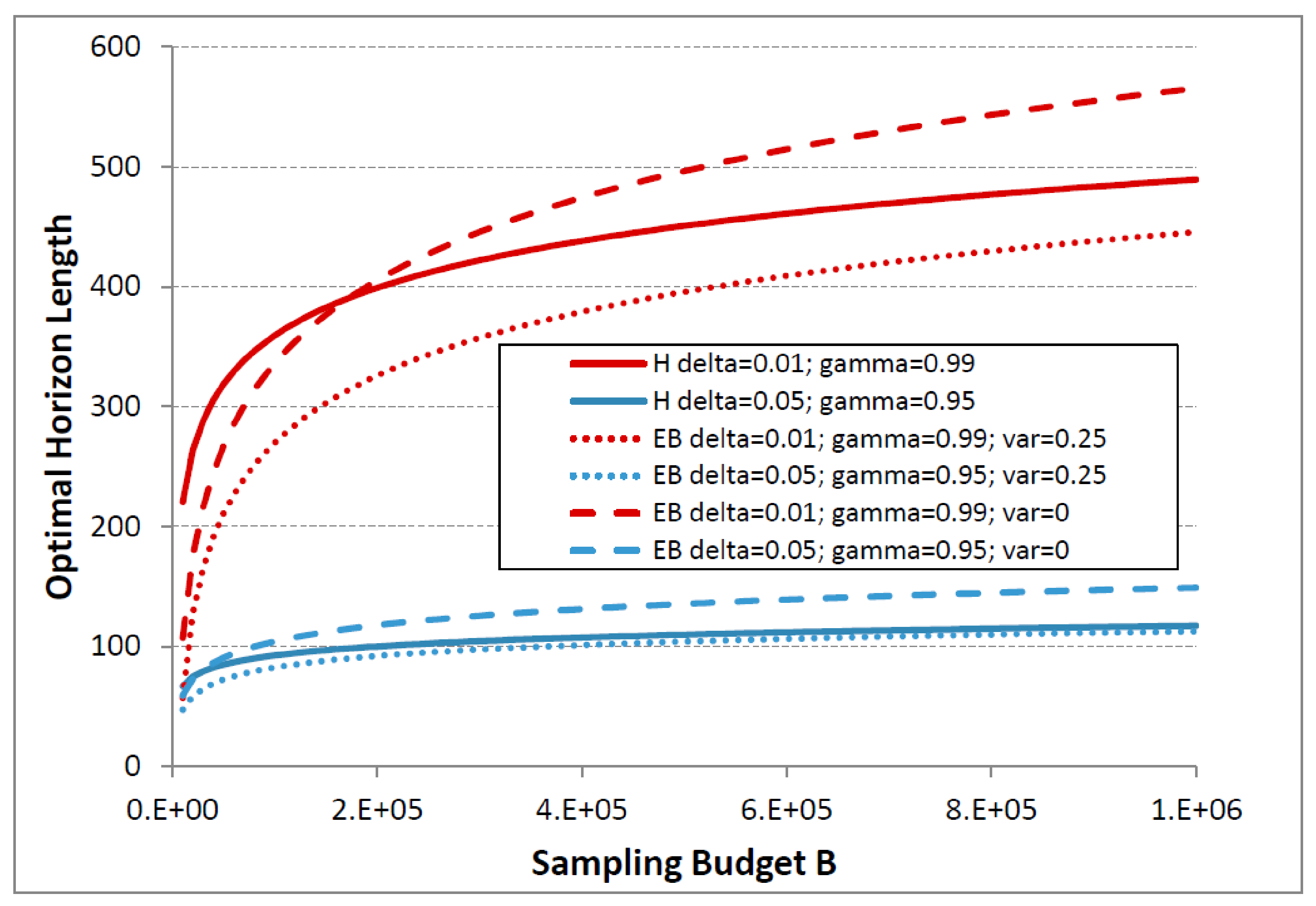
Given a fixed budget B how should trials be allocated?

- For global methods, the only question is the sampling horizon H
- There is no closed form, but H can be determined by solving a simple iteration
- Example: For global Hoeffding bound method:

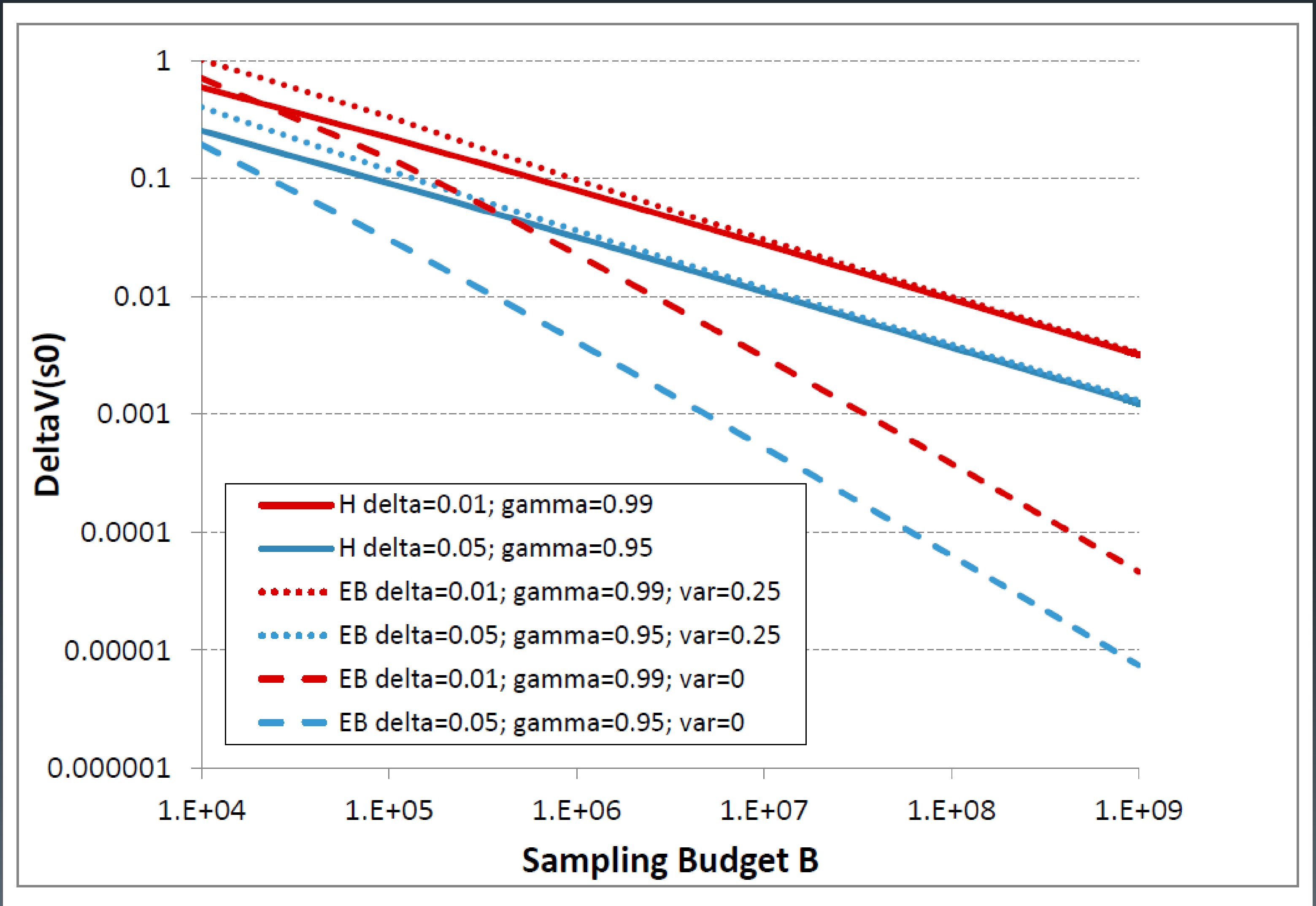
$$H = \frac{\frac{1}{2}\ln\ln\frac{2}{\delta} - \frac{1}{2}\ln 2B - \ln\ln\frac{1}{\lambda}}{\ln\lambda} - \frac{\ln H}{2\ln\lambda}$$

- Similar but more complex iteration for EBB

Optimal Horizon H



Width of the confidence interval for the starting state $\Delta V(s_0)$; $[V_{max} = 1]$



Allocation of Samples for Extended Value Iteration Methods: LCVI(H)

Let $\mu^{\pi}(s)$ be the occupancy measure

$$\mu^{\pi}(s) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^{t} \mathbb{I}[s_i = s] \,\middle|\, s_0, \pi\right]$$

Theorem. $N(s)$ samples should be allocated to state s to minimize

$$\Delta V(s_0) = \sum_{s} \mu(s)^{\pi} 2\gamma V_{max} \sqrt{\frac{\ln 2/\delta}{2N(s)}}$$

Lemma: $N(s)$ samples should be allocated in proportion to $\mu^{\pi}(s)^{2/3}$

It is interesting that more samples are allocated at deeper states than for the global (trajectory-wise) methods, which allocate according to $\mu^{\pi}(s)$.

Allocation of Samples for LCVI(B)

Samples should be allocated to minimize

$$\Delta V(s_0) = \sum_s \mu(s) \left[\frac{\sqrt{c_1 \overline{Var}(s) + \sqrt{c_1 \underline{Var}(s)}}}{\sqrt{N(s)}} + \frac{2c_2}{N(s)} \right]$$

where

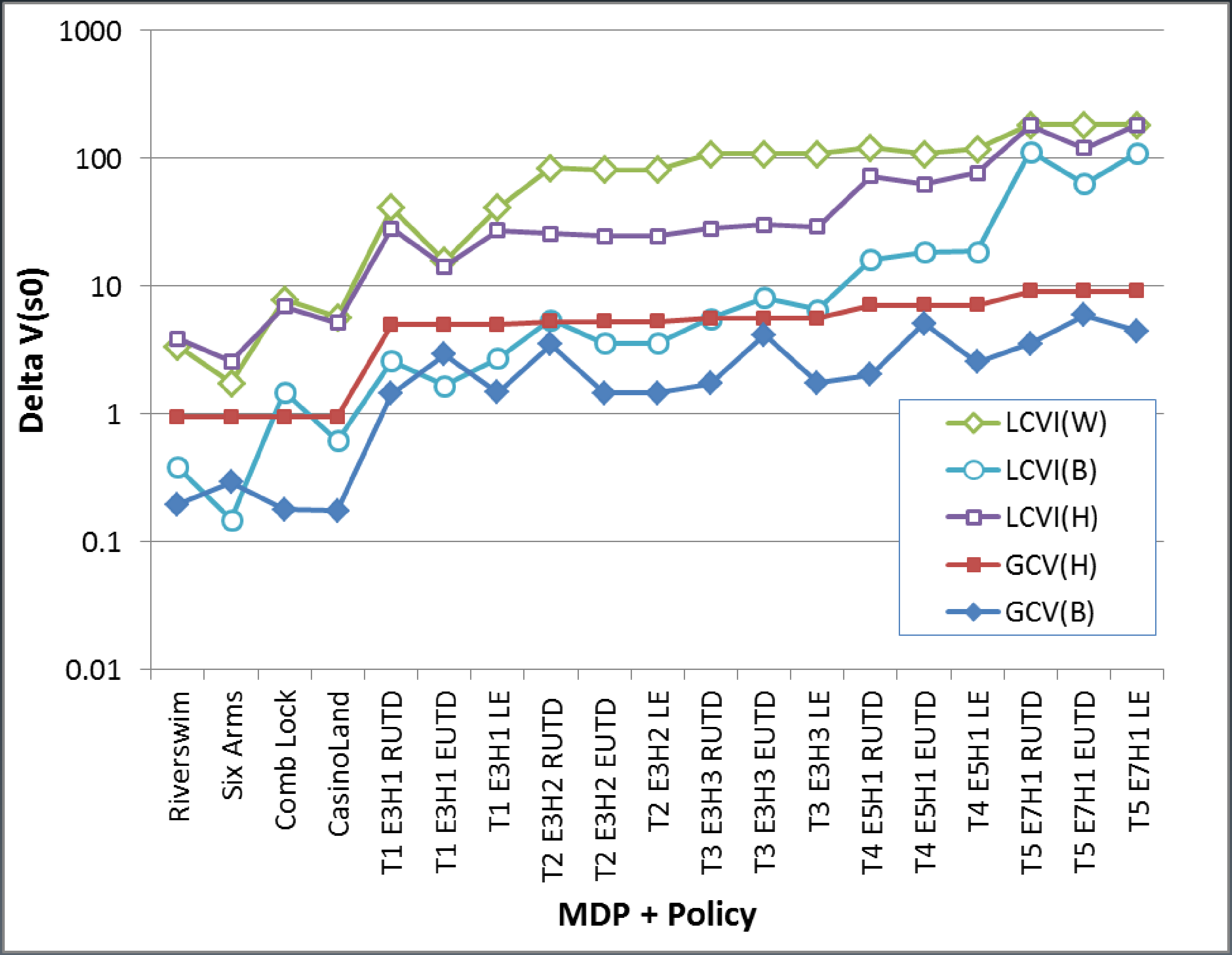
- $c_1 = 2 \ln 3|S|/\delta$ and $c_2 = 3\gamma V_{max} \ln 3/\delta$
- $\overline{Var}(s)$ is an upper bound on the variance of the return at s
- $\underline{Var}(s)$ is a lower bound on the variance of the return at s
- These can be computed via Extended VI

Experimental Comparison

MDP	Policy			Notes
Riverswim	Optimal			
Six Arms	Suboptimal			
Comb Lock	Optimal			some intermediate rewards
CasinoLand	Optimal			added stochasticity
	Edges	Slots		Policies
Tamarisk	3	1		
Tamarisk	3	2		Restore upstream first
Tamarisk	3	3	×	Eradicate upstream first
Tamarisk	5	1		Eradicate leading edge
Tamarisk	7	1		

Policy Evaluation: Results

$$\delta = 0.05; \gamma = 0.95; B = 500{,}000$$



Global Bernstein is almost always best

Local Bernstein wins twice and is by far the best local method

Policy Optimization

- Idea: Use trajectory-based confidence intervals to gain efficiency
- Challenge 1: As we optimize, the policy changes.
  - How can we compute trajectory-based confidence intervals using samples generated from previous policies?
  - Solution: Equivalent Trajectory Method
- Challenge 2: To perform policy improvement, we need to compute $Q_{upper}(s,a)$ for off-policy actions a.
  - This requires local upper confidence limits for each $Q(s,a)$
  - Solution: Use local (extended value iteration) methods for $Q_{upper}(s,a)$ and use a trajectory bound for $V_{lower}(s_0)$
- Result: The Local-Global Confidence Value algorithm (LGCV)

Policy Optimization

- Local-Global Confidence Value (LGCV) algorithm
- Repeat:
  - Draw a minibatch of samples to reduce $V_{upper}(s_0)$ and/or increase $V_{lower}(s_0)$
  - Compute $Q_{upper}(s, a)$ via extended value iteration (EBB)
  - Compute $\pi^{UCB}(s) \coloneqq \arg\max_a Q_{upper}(s, a) \;\; \forall s$
  - Compute $V_{lower}^{\pi^{UCB}}(s_0)$ via a trajectory-wise bound using equivalent trajectories
  - Terminate when

$$\Delta V(s_0) = V_{upper}(s_0) - V_{lower}(s_0) \leq 0.1 \times R_{max}$$

Equivalent Trajectories

- Given:
  - a set of previously-drawn samples $\{N(s,a)\}$ for states $s \in S$ and actions $a \in A$
  - a policy π
- Find:
  - a horizon H
  - an equivalent number of trajectories T
  - such that a trajectory-wise confidence interval is valid

Thought Experiment

- Select H (somehow)
- Estimate $\hat{P}(s'|s,\pi(s))$ from the samples
- Set $M(s,\pi(s)) \coloneqq N(s,\pi(s))$ for all s
- Set $T = 0$ the number of trajectories
- Repeat until $M(s,\pi(s)) = 0$
  - $s \coloneqq s_0$
  - $h \coloneqq 0$
  - while $h < H$
    - $s' \sim \hat{P}$ … draw a sample
    - … $\coloneqq M(s,\pi(s)) - 1$
    - … $(M(s,\pi(s)) < 0)$ return(T)
    - $s \coloneqq s'; h \coloneqq h + 1$
  - $T \coloneqq T + 1$

Too expensive. Let's compute $\mathbb{E}[T]$ instead

Computed $\mathbb{E}[T]$ via stratified MDP

- Select H
- Define an unrolled MDP
  - states: (s, h) for $s \in S$ and $h \in \{1, \dots, H\}$
  - actions: $a \in A$
  - transitions $P((s', h+1)|(s, h), a) = P(s'|s, a)$
  - rewards $R\big((s, h), a\big) = R(s, a)$

- Define $\rho^{\pi}(s, h)$ to be the *undiscounted* occupancy measure for this MDP

Equivalent Number of Trajectories

- Let $Z^\pi(s)$ be the expected number of visits to state s under policy π for trajectories of length H

$$Z^\pi(s) = \sum_{h=0}^{H-1} \rho^\pi(s,h)$$

  Easily computed by dynamic programming along with V^π and the variance Var^π

- Let the equivalent number of trajectories be

$$T^\pi = \min_s \frac{N\big(s, \pi(s)\big)}{Z^\pi(s)}$$

  s is the state that gives the tightest constraint on the number of trajectories

- Claim: $T^\pi = \mathbb{E}[T]$

Computing the Horizon H

- Let $H_{max} = \log_\gamma \frac{\epsilon(1-\gamma)}{2R_{max}}$ (the "ϵ horizon time")

- Choose the H in $\{1, \ldots, H_{max}\}$ that maximizes the "equivalent budget"

$$B_e(H) = HT^\pi(H)$$

- This can be done efficiently by starting with $H = H_{max}$ and working downwards

LGCV is PAC-RL

Simultaneously, with probability at least $1 - \delta$

- $V^*(s_0) \leq V^\pi_{upper}(s_0)$
- $V^\pi_{lower}(s_0) \leq V^*(s_0)$
- $V^\pi_{upper}(s_0) - V^\pi_{lower}(s_0) \leq \epsilon$ by construction

- We employ the Even-Dar et al. trick of using $\delta_t \coloneqq \frac{\delta}{t(t+1)}$ when calculating the t-th confidence interval.

Sample Allocation (Exploration)

- Collect a series of minibatches of size MB
- Let $N = \sum_s N\big(s, \pi(s)\big)$
- Choose Local Sampling vs. Global Sampling
  - Local Sampling:

$$N_{local}(s) = \frac{\mu^{\pi}(s)^{2/3}}{\sum_{s'} \mu^{\pi}(s')^{2/3}}[N + MB]$$

$$N_{local}^{new}(s) = \Big[N_{local}(s) - N\big(s, \pi(s)\big)\Big]_+$$

  - Global Sampling:

$$N_{global}(s) = \frac{\rho^{\pi}(s)}{\sum_{s'} \rho^{\pi}(s')}[N + MB]$$

$$N_{global}^{new}(s) = \Big[N_{global}(s) - N\big(s, \pi(s)\big)\Big]_+$$

Local vs. Global Exploration

Choose the exploration method (local vs. global) that most efficiently shrinks the confidence interval $\Delta V(s_0)$.

"efficiency" = expected improvement per sample

- Local sampling: $\Delta\Delta V_{local}(s_0)$
  - Use extended VI EBB formula assuming no change in variances
- Global sampling: $\Delta\Delta V_{global}(s_0)$
  - Use trajectory-wise EBB formula assuming no change in variances

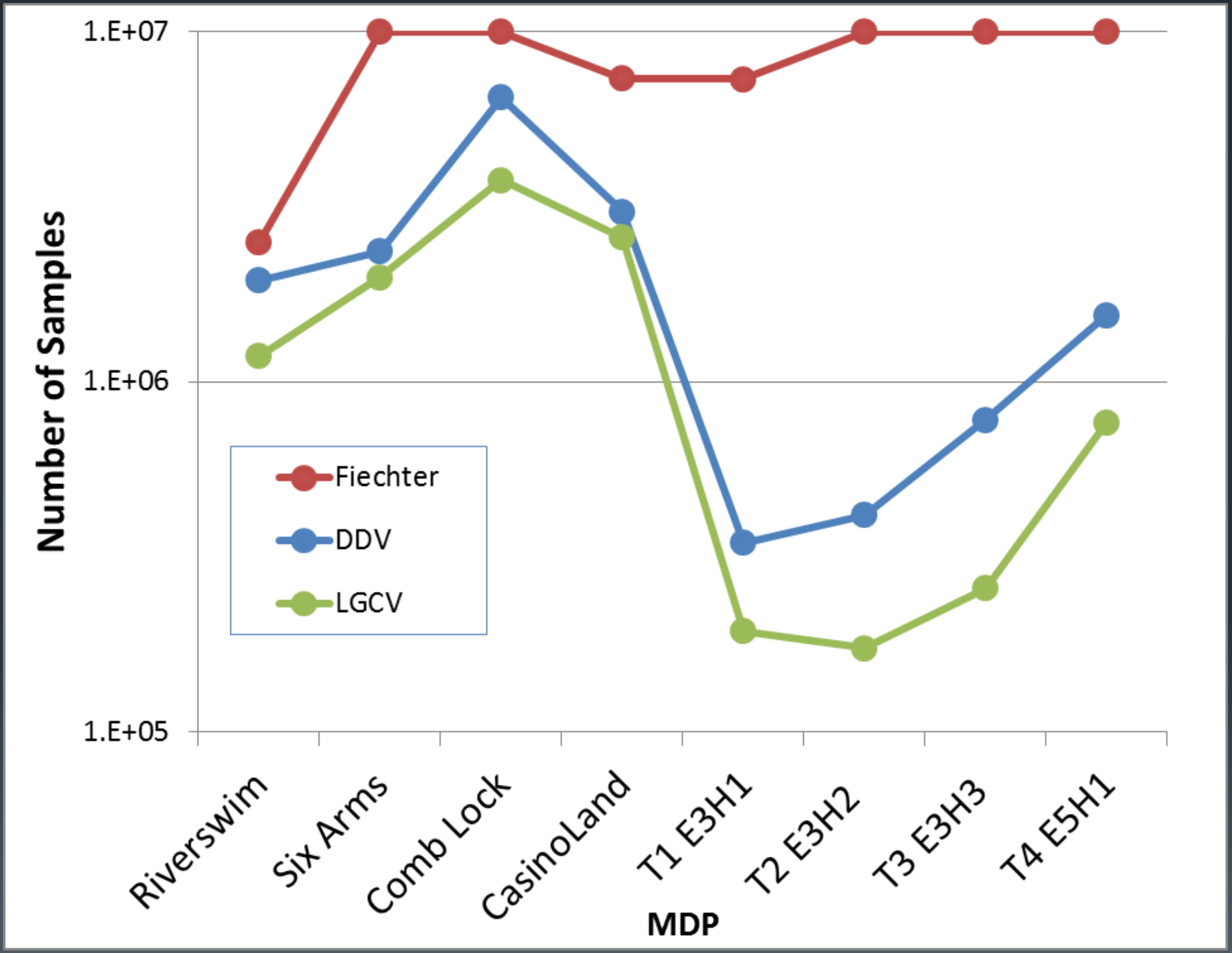
$$\text{Efficiency}_{local} = \frac{\Delta\Delta V_{local}(s_0)}{\sum_s N_{local}^{new}(s)}$$

$$\text{Efficiency}_{global} = \frac{\Delta\Delta V_{global}(s_0)}{\sum_s N_{global}^{new}(s)}$$

Policy Optimization Experiments

- Methods:
  - Fiechter: Samples along trajectories to maximize the total shrinkage of local Hoeffding confidence intervals (Fiechter, 1994)
  - DDV: Local Extended Value Iteration with EBB to greedily reduce $\Delta V(s_0)$. Extends (Dietterich, Taleghan & Crowley, 2013)
  - LGCV: Our new method
- Metric: # of samples required to drive $\Delta \mathrm{V}(s_0) \leq 0.1 \times R_{max}$ with probability 0.95
  - Halted at 1×10^7 samples

Policy Optimization Results



Summary

- New algorithms for Monte Carlo policy evaluation
- Experiments show that in our benchmark problems, the Empirical Bernstein Bound is tighter than Hoeffding or Weissman
  - Trajectory-wise EBB is usually tighter than the bound obtained by Extended Value Iteration using a local EBB at each state
- New PAC-RL algorithm for MDP planning
  - Combines an upper bound based on EVI with local EBB
  - And a lower bound based on equivalent trajectories and global EBB