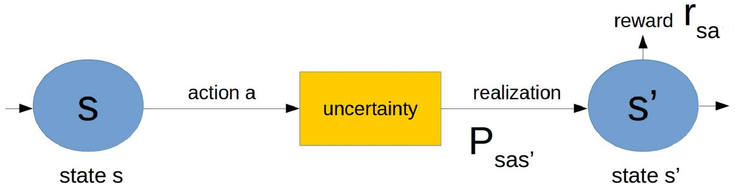
A First-Order Approach to Accelerated Value Iteration

Vineet Goyal, Julien Grand-Clement

{vg2277, jg3728}@columbia.edu, Columbia University

Markov model.

Markov Decision Process: sequential decision making.



Goal: compute optimal sequence of decisions.
There are n states and a discount factor of $\lambda \in (0, 1)$.
Applications: reinforcement learning, healthcare, dynamic pricing, inventory planning, etc.

Classical algorithms.

- *Bellman operator* $T : \mathbb{R}^n \to \mathbb{R}^n$, where for $\boldsymbol{v} \in \mathbb{R}^n$,
$$T(\boldsymbol{v})_i = \max_{\pi} \Sigma_a \, \pi_{ia} \cdot \left(r_{ia} + \lambda \cdot \boldsymbol{P}_{ia}^{\top} \boldsymbol{v}\right).$$
Goal: find $\boldsymbol{v}^*$ such that $T(\boldsymbol{v}^*) = \boldsymbol{v}^*$.
- Value iteration (VI):
$$\boldsymbol{v}_0 \in \mathbb{R}^n, \boldsymbol{v}_{s+1} = T(\boldsymbol{v}_s), \forall\, s \geq 0. \qquad \text{(VI)}$$
$\to \boldsymbol{v}_s = \boldsymbol{v}^* + \mathcal{O}\left(\lambda^s\right)$ (T is λ-contraction),
$\to$ Problem: running time scales as $\approx 1/(1-\lambda)$.

Value Iteration and Gradient Descent.

- Gradient Descent:
$$\boldsymbol{v}_0 \in \mathbb{R}^n, \boldsymbol{v}_{s+1} = \boldsymbol{v}_s - \alpha_s \nabla f(\boldsymbol{v}_s), \forall\, s \geq 0. \qquad \text{(GD)}$$
- Idea: $\boldsymbol{v} - T(\boldsymbol{v}) = (\boldsymbol{I} - T)(\boldsymbol{v})$ = gradient of $f : \mathbb{R}^n \to \mathbb{R}^n$.
- New algorithm:
$$\boxed{\boldsymbol{v}_0 \in \mathbb{R}^n, \boldsymbol{v}_{s+1} = \boldsymbol{v}_s - \alpha_s \left(\boldsymbol{v}_s - T(\boldsymbol{v}_s)\right), \forall\, s \geq 0.} \qquad \text{(R-VI)}$$
Recall for μ-strongly convex, L-smooth function f:
$$\mu \leq \frac{\|\nabla f(\boldsymbol{v}) - \nabla f(\boldsymbol{w})\|_2}{\|\boldsymbol{v} - \boldsymbol{w}\|_2} \leq L. \qquad (1)$$
For our Bellman operator T,
$$(1-\lambda) \leq \frac{\|(\boldsymbol{I} - T)(\boldsymbol{v}) - (\boldsymbol{I} - T)(\boldsymbol{w})\|_\infty}{\|\boldsymbol{v} - \boldsymbol{w}\|_\infty} \leq (1+\lambda). \qquad (2)$$

Main insights.

- Connection: Convex Optimization and Value Iteration.
- $\boldsymbol{I} - T$ is the gradient of some function.
- Difficulty: we work with $\|\cdot\|_\infty$ instead of $\|\cdot\|_2$.
- New algorithm: Accelerated Value Iteration (A-VI).
$$\boldsymbol{v}_0, \boldsymbol{v}_1 \in \mathbb{R}^n, \begin{cases} \boldsymbol{h}_s = \boldsymbol{v}_s + \gamma_s \cdot (\boldsymbol{v}_s - \boldsymbol{v}_{s-1}), \\ \boldsymbol{v}_{s+1} \leftarrow \boldsymbol{h}_s - \alpha_s \left(\boldsymbol{h}_s - T\left(\boldsymbol{h}_s\right)\right), \end{cases} \forall\, s \geq 1.$$

Our results.

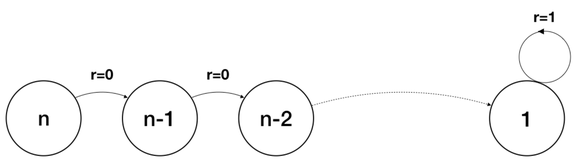
We tune step sizes using (1) and (2).

1. R-VI converges, and running time scales as $\approx 1/(1-\lambda)$.

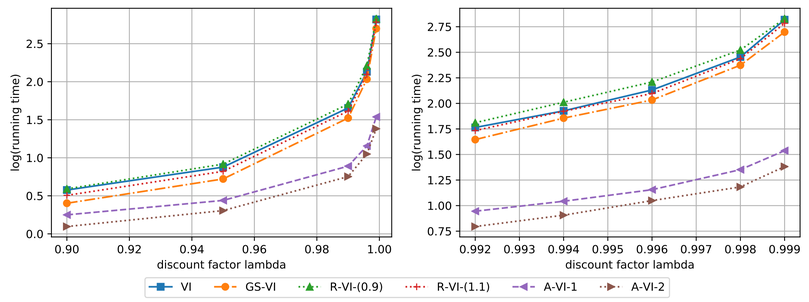
2. Suppose $T = T^\pi$ (affine operator).
$\to$ If the MDP is *reversible*, A-VI converges and running time scales as $\approx 1/\sqrt{1-\lambda}$.
$\to$ Without reversibility, A-VI may diverge (ex.: cycle).

3. Suppose $T =$ Bellman operator.
$\to$ A-VI is equivalent to Linear Time-Varying Dynamical System.
$\to$ Key role of the joint spectral radius of a set of matrices: a hard problem!

4. First-order algorithm:
$$\boldsymbol{v}_0 = \boldsymbol{0}, \boldsymbol{v}_{s+1} \in span\{\boldsymbol{v}_0, ..., \boldsymbol{v}_s, T(\boldsymbol{v}_0), ...T(\boldsymbol{v}_s)\}, s \geq 0.$$
Theorem. There exists a hard MDP instance such that no first order algorithm can converge faster than Value Iteration, during the first n iterations.
$\to$ Key difference with convex optimization!
$\to$ Our hard MDP instance:



Numerical experiments.



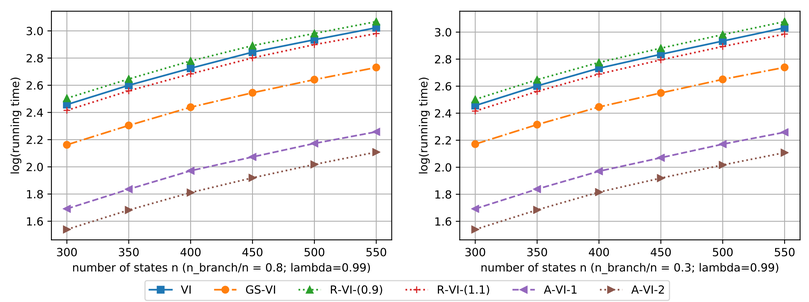


Figure 1: Running time of A-VI vs. state-of-the-art algorithms (log-scale).

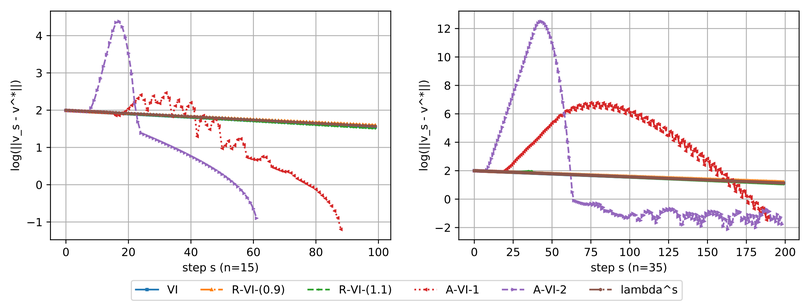


Figure 2: Running time of A-VI and VI on our hard MDP instance.

Reference: *A First-Order Approach to Accelerated Value Iteration*, V. Goyal and J. Grand-Clement, submitted, https://arxiv.org/abs/1905.09963.
Webpage: *http : //www.columbia.edu/ ∼ jg3728/*