n-Step Temporal Difference Learning with an Optimal n

Shalabh Bhatnagar

August 14, 2025

Department of Computer Science and Automation
& Robert Bosch Centre for Cyber Physical Systems
Indian Institute of Science
Bangalore 560012

Key Takeaways[1]

- TD learning is an incremental-update RL algorithm for prediction
- n-step TD learning tries to balance Monte-Carlo approaches with 1-step TD by having a variable n
- Important Observation: Different n-values result in different RMSE
- Question we address: How to adaptively select n in n-step TD
- Algorithm used: Two-timescale Discrete SPSA with n-step TD
- Main Results:
    - Prove almost sure convergence of the resulting two-timescale scheme using a differential inclusions analysis
    - Demonstrate experimentally that the scheme gives optimal RMSE and is better than the well-known OCBA procedure for discrete stochastic optimization

[1] L.Mandal and S.Bhatnagar, n-step temporal difference learning with an optimal n, Automatica, Article 112449, 2025; Arxiv: https://arxiv.org/pdf/2303.07068, 2024.

Outline of the Talk

1. Markov decision processes and RL
2. The prediction problem and Monte-Carlo approaches
3. Stochastic approximation and model-free approaches
4. Stochastic optimization and SPSA
5. One-simulation SPSA with the smallest cyclic cancellation of bias
6. TD learning and *n*-step TD learning
7. Discrete random projections
8. Differential inclusions based analysis under lack of Lipschitz continuity of the objective
9. Key experiments and comparisons with OCBA

Markov Decision Processes[2]

- Consider a sequence of random variables (MDP) $\{X_n\}$, $X_n \in S$, $\forall n$, that depends on a control-valued sequence $\{Z_n\}$, $Z_n \in A$, $\forall n$, and which satisfies the controlled Markov property.
- Here $S \equiv$ state Space and $A \equiv$ action space.
- Assume S and A are finite sets.
- Let $k(X_n, Z_n, X_{n+1})$ be the cost incurred when state at time n is X_n, action chosen is Z_n and the next state is X_{n+1}.

[2] M.L.Puterman, Markov Decision Processes, John Wiley, 1995

The Controlled Markov Property

- For all $i_0, i_1, \ldots, s, s', b_0, b_1 \ldots, a$ in appropriate sets,

$$P(X_{n+1} = s' \mid X_n = s, Z_n = a, \ldots, X_0 = i_0, Z_0 = b_0)$$
$$= P(X_{n+1} = s' \mid X_n = s, Z_n = a) = p(s, s', a)$$

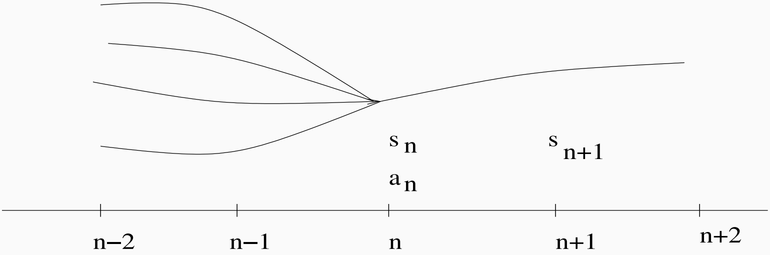


Figure 1: The Controlled Markov Behaviour

The Infinite Horizon Discounted Cost Problem

- Objective: Find a sequence of controls $\{Z_n\}$ that minimizes the cost-to-go or the value function

$$V_{\{Z_n\}}(i) = E\left[\sum_{j=0}^{\infty} \gamma^j k(X_j, Z_j, X_{j+1}) \mid X_0 = i\right]$$

- Let $V^*(i) = \min_{\{Z_n\}} V_{\{Z_n\}}(i)$
- The Bellman equation: The optimal cost function V^* satisfies

$$V^*(i) = \min_{a \in A(i)} \sum_j p(i,j,a)(k(i,a,j) + \gamma V^*(j)), \quad i \in S.$$

Further, V^* is the unique solution of this equation within the class of bounded functions.

The Prediction Problem

- By a policy, we mean a sequence of functions $\{\pi_0, \pi_1, \ldots\}$ with $\pi_i : S \to A$, $i = 0, 1, \ldots$
- A stationary policy π is one where $\pi_i = \pi_j \equiv \pi$, $\forall i \neq j$.
- The Prediction Problem Given a policy π, find it's value $V_\pi(s)$ where

$$V_\pi(s) = E_\pi \left[\sum_{j=0}^{\infty} \gamma^j k(X_j, Z_j, X_{j+1}) \mid X_0 = i \right]$$

- Bellman Equation for Policy π

$$V_\pi(i) = \sum_j p(i, j, \pi(i))(k(i, \pi(i), j) + \gamma V_\pi(j)), \quad i \in S.$$

The Reinforcement Learning Setting

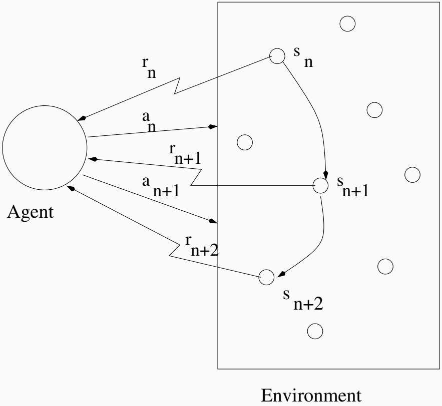


Figure 2: Agent-Environment Interaction

- No access to transition probabilities but data.
- Often single-stage rewards (possibly random) in place of costs.

Monte-Carlo Based Prediction

- Recall that $V_\pi(i) = E_\pi\left[\sum_{j=0}^{\infty} \gamma^j r_j | X_0 = i\right]$, $i \in S$.
- Monte-Carlo Estimates of $V_\pi(i)$: Run multiple episodes with policy π.
  - Episode k: $s_0^k, a_0^k, r_0^k, s_1^k, a_1^k, r_1^k, s_2^k, \ldots, s_{T^k-1}^k, a_{T^k-1}^k, r_{T^k-1}^k, s_{T^k}^k$.
  - Assume state i is visited N times.
  - Let return from the mth visit to state i, visited at instant l, be defined as

$$G_l^m(i) = \sum_{t=0}^{T^m-l-1} \gamma^t r_{t+l}^m$$

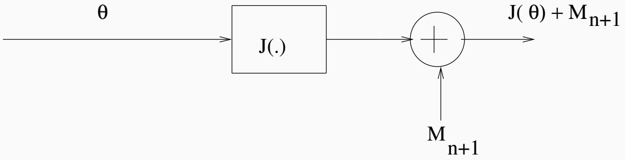
- Monte-Carlo estimate of $V_\pi(i)$

$$\hat{V}_\pi(i) = \frac{1}{N} \sum_{m=1}^{N} G_l^m(i).$$

- Problem with Monte-Carlo: Cannot start updates unless a full episode is run.

Stochastic Approximation[3]

- Objective: Solve the equation $J(\theta) = 0$ when analytical form of J is not known, however, 'noisy' measurements $J(\theta(n)) + M_{n+1}$ can be obtained



- The Robbins-Monro Algorithm:

$$\theta(n+1) = \theta(n) + a(n)(J(\theta(n)) + M_{n+1}) \tag{1}$$

[3] H.Robbins and S.Monro *Annals of Mathematical Statistics*, 22: 400–407, 1951

Applications of SA[4]

- Convergence of SA can be shown under fairly general assumptions
- Applications
  - Noisy fixed Point Computation – Find θ^* s.t. $f(\theta^*) = \theta^*$ under noisy measurements of f

$$J(\theta) = f(\theta) - \theta$$

  - Noisy gradient scheme – Find local minima of f

$$J(\theta) = -\nabla f(\theta)$$

[4] V.S.Borkar, Stochastic Approximation: A Dynamical Systems Viewpoint, Hindustan Book Agency, 2022.

A General Convergence Result[5] [6]

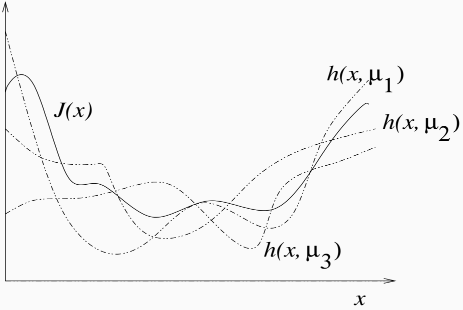
- (C1) $J : \mathcal{R}^N \to \mathcal{R}^N$ is Lipschitz continuous
- (C2) $\sum_n a(n) = \infty, \ \sum_n a(n)^2 < \infty$
- (C3) $M_{n+1}, n \geq 0$ is a martingale difference w.r.t. $\{\mathcal{F}_n\}$, where $\mathcal{F}_n = \sigma(\theta(m), M_m, m \leq n), n \geq 1$. Further, for some $K > 0$,
$$E[\| M_{n+1} \|^2 | \mathcal{F}_n] \leq K(1 + \| \theta(n) \|^2)$$
- (C4) $\sup_n \| \theta(n) \| < \infty$ almost surely
- Theorem Under (C1)-(C4), $\theta(n) \to A$ a.s., where A is a (sample-path dependent) compact connected internally chain recurrent set of the ODE $\dot{\theta} = h(\theta)$.

[5] M.Benaim, A dynamical system approach to stochastic approximations, SIAM J.Contr.Optim., 34(2):437-472, 1996.

[6] V.S.Borkar, Stochastic Approximation: A Dynamical Systems Viewpoint, Hindustan Book Agency, 2022.

A Problem of Stochastic Optimization

- Let $J : \mathcal{R}^N \to \mathcal{R}$ be a given objective function having the form $J(x) = E_\mu[h(x, \mu)]$, where μ denotes 'noise' and $E_\mu[\cdot]$ is the expectation under that noise



- $\boxed{\text{AIM: Find } x^* \text{ s.t. } J(x^*) = \min_{x \in \mathcal{R}^N} J(x)}$

Gradient Estimation Schemes[7]

- Single-simulation classical perturbation analysis schemes based on sample performance gradients: require $\nabla J(x) = \nabla E_\mu[h(x,\mu)] = E[\nabla_\mu h(x,\mu)]$.
- Zeroth-order gradient estimation methods
  - Finite-Difference Stochastic Approximation - Kiefer & Wolfowitz (1952): require $2N$ simulations for one gradient estimate
  - Random Perturbation Approaches
    - SPSA: one or two simulations with Bernoulli perturbation
    - SF: one or two simulations with Gaussian or Cauchy perturbations
    - RDSA: one or two simulations with uniform on the hyper-rectangle
- Where applicable direct gradient schemes are the best but many times they are not applicable.

[7] S.Bhatnagar, H.L.Prasad and L.A.Prashanth, Stochastic Recursive Algorithms for Optimization: Simultaneous Perturbation Methods, Springer, 2013.

Simultaneous Perturbation Stochastic Approximation

- Let $\Delta(n) = (\Delta_1(n), \ldots, \Delta_N(n))^T$ be a vector of i.i.d., ± 1-symmetric, Bernoulli random variables.
- Two-simulation SPSA estimate:[8] Run two simulations with parameters $\theta(n) + \delta\Delta(n)$ and $\theta(n) - \delta\Delta(n)$.

$$\tilde{\nabla}_i J(\theta) = (h(\theta + \delta\Delta, \mu_1) - h(\theta - \delta\Delta), \mu_2))/2\delta\Delta_i.$$

- One-simulation SPSA estimate:[9] Run one simulation with parameter $\theta(n) + \delta\Delta(n)$.

$$\tilde{\nabla}_i J(\theta) = h(\theta + \delta\Delta, \mu_1)/\delta\Delta_i.$$

[8] J.C.Spall, *IEEE Transactions on Automatic Control*, 37(3):332-341,1992.
[9] J.C.Spall, *Automatica*, 33(1):109-112, 1997.

Consistency of the SPSA Estimators

- Two-Simulation Estimator: Using Taylor's expansions,

$$E_{\theta(n)}\left[\frac{J(\theta(n)+\delta\Delta(n))-J(\theta(n)-\delta\Delta(n))}{2\delta\Delta_i(n)}\right]$$

$$=E_{\theta(n)}\left[\frac{\Delta(n)^T\nabla J(\theta(n))}{\Delta_i(n)}\right]+o(\delta)=\nabla_i J(\theta(n))+o(\delta).$$

- One-Simulation Estimator: Using a Taylor's expansion,

$$E_{\theta(n)}\left[\frac{J(\theta(n)+\delta\Delta(n))}{\delta\Delta_i(n)}\right]=E_{\theta(n)}\left[\frac{J(\theta(n)}{\delta\Delta_i(n)}\right]$$

$$+E_{\theta(n)}\left[\frac{\Delta(n)^T\nabla J(\theta(n))}{\Delta_i(n)}\right]+O(\delta)=\nabla_i J(\theta(n))+O(\delta).$$

Problem with One-Simulation SPSA and Alternative

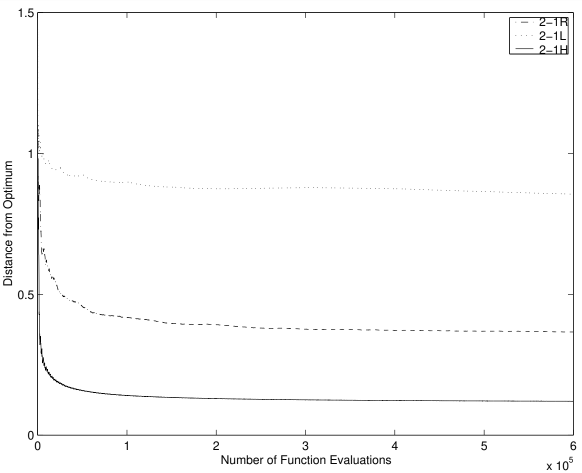
- Both one and two simulation estimators give an approximate gradient direction.
- Even though one-simulation SPSA is desirable in "real-world" scenarios, it suffers from an additional bias term resulting in poor performance.
- Alternative: Use One-SPSA but with ± 1-valued deterministic perturbations (instead of randomized) that cancel cyclically at regular intervals.[10][11]

---

[10] S.Bhatnagar, M.Fu, S.Marcus and I.Wang, ACM Transactions on Modeling and Computer Simulation, 13(2):180-209, 2003.

[11] S.Bhatnagar, H.L.Prasad and L.A.Prashanth, Stochastic Recursive Algorithms for Optimization: Simultaneous Perturbation Methods, Springer, 2013.

One-Simulation Deterministic Perturbation SPSA vs. Randomized SPSA [Bhatnagar et al. (2003)]



Temporal Difference Learning

- Recall Bellman equation for a given policy π:

$$V_\pi(i) = \sum_{j \in S} p(i, \pi(i), j)(r(i, \pi(i), j) + \gamma V_\pi(j)), \ i \in S.$$

- TD performs incremental updates using stochastic approximation.
- Let $V_n = (V_n(1), \ldots, V_n(|S|))^T$ and $s_n =$ state visited at time n.
- TD update incorporates bootstrapping: $\forall n$,

$$V_{n+1}(s_n) = V_n(s_n) + a(n)(r_n + \gamma V_n(s_{n+1}) - V_n(s_n)),$$

with $V_{n+1}(j) = V_n(j), \forall j \neq s_n$.

n-Step TD Learning

- n-step Bellman equation (for a given policy):

$$V_\pi(i_0) = \sum_{k=0}^{n-1} p(i_k, \pi(i_k), i_{k+1})(\gamma^k r(i_k, \pi(i_k), i_{k+1}) + \gamma^n V_\pi(i_n)).$$

- n-step TD is a bridge between TD and Monte-Carlo.
- Let $\hat{V}^n(S_\kappa) = \sum_{j=\kappa+1}^{\min(\kappa+n,T)} \gamma^{j-\kappa-1} R_j$.
- n-step TD update: If $\kappa + n < T$,

$$\hat{V}^n(S_\kappa) := \hat{V}_n(S_\kappa) + \gamma^n V_n(S_{\kappa+n}),$$

$$V_{n+1}(S_\kappa) = V_n(S_\kappa) + a(n)[\hat{V}_n(S_\kappa) - V_n(S_\kappa)].$$

What n to use in n-step TD

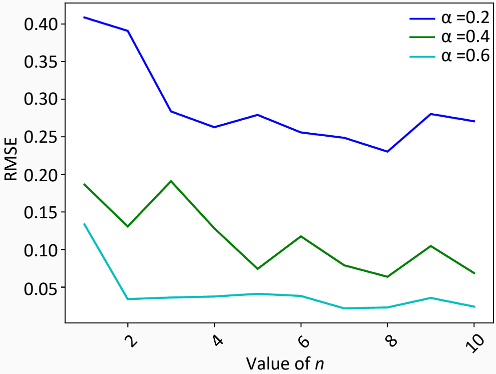


Figure 3: Obtained RMSE for different values of n and fixed α on a Random Walk Example.

- Different values of n give rise to different estimator variance.
- Objective: Find n that adaptively minimizes RMSE.

MSE and Projection to the Convex Hull

- Let $g_n(S_i) \stackrel{\triangle}{=} (\hat{V}^n(S_i) - V_n(S_i))^2$.
- Goal: Find $n^* \in D = \{1, 2, \ldots, L\}$ that minimizes the long-run average MSE

$$J(n) = \lim_{m \to \infty} \frac{1}{m} \mathbb{E}\left[\sum_{i=1}^{m} g_n(S_i)\right].$$

- Let $\bar{D} = [1, L] \stackrel{\triangle}{=}$ the closed convex hull of the discrete parameter set D.
- Let $\bar{\Gamma} : \mathbb{R} \to \bar{D}$ denote the projection

$$\bar{\Gamma}(x) = \min(L, \max(x, 1)),$$

  to the set $\bar{D}$.
- The n-update will proceed in the space $\bar{D}$ but the actual values are decided by a second projection operator.

Random Projection to the Discrete Parameter Space[12]

- For $n \in \mathbb{R}$, let $k \leq n \leq k+1$, $1 \leq k < L$,

$$\Gamma(n) := \begin{cases} k, & \text{w.p. } (k+1-n) \\ k+1, & \text{w.p. } (n-k) \end{cases} \tag{2}$$

  and for $n < 1$ or $n > L$, we let

$$\Gamma(n) := \begin{cases} 1, & \text{if } n < 1 \\ L, & \text{if } n \geq L. \end{cases} \tag{3}$$

- For $k \leq n \leq k+1$ (i.e., $n \in \bar{D}$), let

$$\hat{V}_n(x) := \beta \hat{V}^k(x) + (1-\beta)\hat{V}^{k+1}(x).$$

[12] S. Bhatnagar S, V.K. Mishra, and N. Hemachandra, Stochastic algorithms for discrete parameter simulation optimization, IEEE Transactions on Automation Science and Engineering, 8(4):780-93, 2011.

Parameters in the Algorithm

- Step-Size Sequences: Consider two step-size sequences $\{a_m\}$ and $\{b_m\}$ satisfying the following conditions:

$$a_m, b_m > 0, \ \forall n,$$

$$\sum_m a_m = \sum_m b_m = \infty, \ \frac{a_{k+1}}{a_k} \to 1 \text{ as } k \to \infty,$$

$$\sum_m a_m^2 < \infty, \sum_m b_m^2 < \infty, \ \lim_{m \to \infty} \frac{a_m}{b_m} = 0.$$

- Sensitivity Parameter: Let $\delta > 0$ be a small constant.
- Perturbation Sequence: Define the perturbation sequence $\{\Delta_m\}$ as follows: $\Delta_m = +1$ on even iterations and -1 on odd iterations.

n-Step TD Algorithm with Adaptive n

- Update Equations: For $m \geq 0$, $i \in S$,

$$
\begin{aligned}
n_{m+1} &= \bar{\Gamma}\left(n_m - a_m \frac{Y_{m+1}}{\delta \Delta_m}\right), & (4) \\
Y_{m+1} &= Y_m + b_m \left(g_{n_m^+}(S_m) - Y_m\right), & (5) \\
V_{m+1}(i) &= V_m(i) + b_m I_{S_m}(i)(\hat{V}_{n_m^+}(i) - V_m(i)). & (6)
\end{aligned}
$$

- Here $n_m^+ = \bar{\Gamma}(n_m + \delta \Delta_m)$.
- Also, $g_{n_m^+}(S_m) \stackrel{\triangle}{=} (\hat{V}_{n_m^+}(S_m) - V_m(S_m))^2$.
- Also,

$$
I_{S_m}(i) = \begin{cases} +1 & S_m = i \\ 0 & \text{otherwise,} \end{cases}
$$

accounts for asynchronous updates.

Lack of Regularity at $k \in D$

- Lemma 1: $J(n)$ is a Lipschitz continuous function in $n \in \bar{D}$. Further, its derivative is piecewise Lipschitz continuous on intervals $[k, k+1)$, $1 \leq k \leq L$ but discontinuous in general with points of discontinuity in the set D.
- We obtain in particular that

$$\left| \frac{dJ(n)}{dn}|_{n=m} - \frac{dJ(n)}{dn}|_{n=l} \right| \leq K_1 |m - l|,$$

for all $m, l \in [k, k+1)$,$1 \leq k \leq L$.

- However,

$$\lim_{n \downarrow k} \frac{dJ(n)}{dn} \neq \lim_{n \uparrow k} \frac{dJ(n)}{dn}.$$

The Faster Timescale Analysis

- Consider the following system of ODEs corresponding to the fast timescale:

$$\begin{aligned} \dot{n}(t) &= 0, & (7) \\ \dot{Y}(t) &= J(\bar{\Gamma}(n(t)+\delta\Delta(t))) - Y(t), & (8) \\ \dot{V}(t) &= \mathbb{D}(V_{n(t)^+}(t) - V(t)). & (9) \end{aligned}$$

- From (7), $n(t) \equiv n$, $\forall t$, hence (8)-(9) become

$$\begin{aligned} \dot{Y}(t) &= J(\bar{\Gamma}(n+\delta\Delta(t))) - Y(t), & (10) \\ \dot{V}(t) &= \mathbb{D}(V_{n^+} - V(t)). & (11) \end{aligned}$$

- Here $\Delta(t) = \Delta_m$, for $t \in [\sum_{i=0}^{m-1} a(i), \sum_{i=0}^{m} a(i)]$, $m \geq 1$.
- Now (10) has $Y^* = \lambda(n) \equiv J(\bar{\Gamma}(n+\delta\Delta_m))$ as its unique GASE.
- $V^* = V_{n^+}$ is the unique GASE of (11).

Discontinuity of Slower Scale ODE

- Proposition 1: The following hold:
  (a) $\| Y_m - J(\bar{\Gamma}(n + \delta\Delta_m))\| \to 0$ a.s. as $m \to \infty$,
  (b) $\| V_m - V_{n^+} \| \to 0$ a.s. as $m \to \infty$.
- Consider now the slower timescale recursion for the n-update. The associated ODE is the following:

$$\dot{n}(t) = \hat{\bar{\Gamma}}\left(-\frac{J(\bar{\Gamma}(n(t) + \delta\Delta(t)))}{\delta\Delta(t)}\right). \tag{12}$$

- If $\dot{J}(n)$ exists and is Lipschitz, then one can argue that $\{n_m\}$ would converge almost surely to a neighborhood of the set of attractors of the ODE

$$\dot{n}(t) = \hat{\bar{\Gamma}}(-\dot{J}(n)).$$

- However, $\dot{J}(n)$ is discontinuous in general for $n \in D$ (Lemma 1).

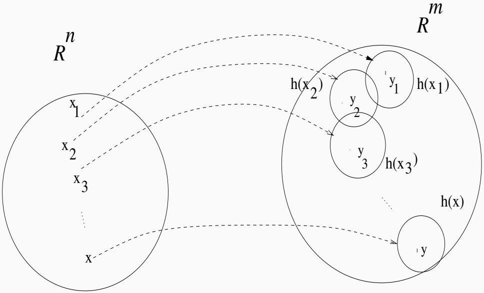
A Set-Valued Map for the Slower Dynamics

- Define a set-valued map $H(n)$ as follows:

$$H(n) = \cap_{\eta>0} \cap \overline{co}(\{\hat{\hat{\Gamma}}(-\dot{J}(m)) | \|m-n\| < \eta\}).$$

- For $n \in (k, k+1)$ with $k, k+1 \in D$, $H(n) = -\dot{J}(n)$
- For $n = k \in [2, L-1]$, $H(n) = [\alpha_k, \beta_k]$, where $\alpha_k \equiv$ lower limit of $\dot{J}(n)$ at $n = k$ and $\beta_k \equiv$ upper limit of $\dot{J}(n)$ at $n = k$.
- For $n = 1$ and $n = L$, we still let $H(n) = [\alpha_k, \beta_k]$ with $k = 1$ or L if $0 \in H(n)$. Else, we take the closed convex hull of the points $0, \alpha_k, \beta_k$ when $k = 1$ or $k = L$.

Marchaud Set-Valued Map



- A set-valued map h is called Marchaud if
  - $h(x)$ is convex and compact for each x
  - $\sup_{w \in h(x)} \| w \| \leq K(1 + \| x \|)$ for each x
  - h is upper-semicontinuous, i.e., given $\{x_n\} \subset \mathcal{R}^n$ and $\{y_n\} \subset \mathcal{R}^m$ with $x_n \to x$ and $y_n \to y$ with $y_n \in h(x_n), \forall n$, we have $y \in h(x)$

Results

- Lemma 2: The set-valued map $H(n)$ is Marchaud.
- Consider the Differential Inclusion

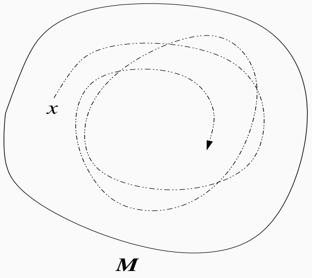
$$\dot{n}(t) \in H(n(t)). \tag{13}$$

- Thus, every solution to the above DI is absolutely continuous.[13]

[13] J. Aubin and A. Cellina, Differential Inclusions: Set-Valued Maps and Viability Theory, Springer, 1984.

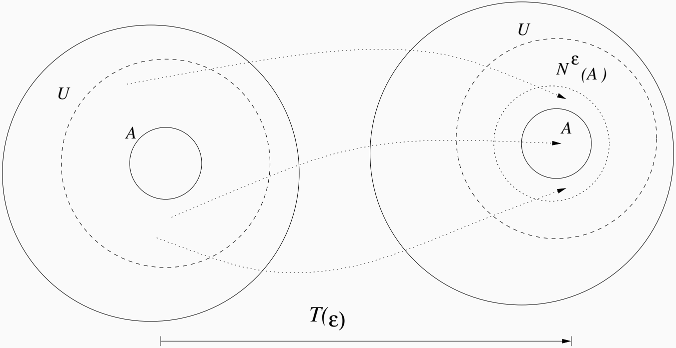
Invariant Set

- $M \subset \mathcal{R}^d$ is invariant if for every $x \in M$, there exists $\mathbf{x} \in \Sigma$ s.t. $x(t) \in M \; \forall t$ with $x(0) = x$



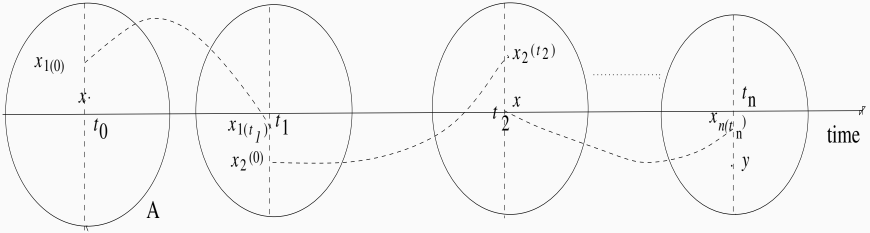
Attractor of a DI

- $A \subset \mathcal{R}^d$ is attracting if it is compact and there exists a neighborhood U such that for any $\epsilon > 0$, $\exists T(\epsilon) \geq 0$ with $\Phi([T(\epsilon), \infty), U) \subset N^\epsilon(A)$



- If the above A is invariant, it is called an attractor

Internally Chain Transitive Set



- We say $x \overset{A}{\to} y$ if $\forall \epsilon, T > 0, \exists n > 0$, solutions $x_1, \dots, x_n$ to DI and time points $t_1, \dots t_n$ with $t_n - t_{n-1} \geq T$, such that
  (a) $x_i(s) \in A, \forall 0 \leq s \leq t_i - t_{i-1}, i = 1, \dots, n$
  (b) $\|x_i(t_i) - x_{i+1}(0)\| \leq \epsilon, \forall i$
  (c) $\|x_1(0) - x\|, \|x_n(t_n) - y\| \leq \epsilon$
- $(x_1, \dots, x_n)$ is called (ϵ, T) chain in A from x to y.
- The set A is ICT if it is compact and $x \overset{A}{\to} y$ for all $x, y \in A$.

Main Result

- Theorem 1: $n_m \to P$ almost surely as $m \to \infty$, where P is an internally chain transitive set of the DI (13).
- Proof: We can rewrite (4) as follows:

$$n_{m+1} = \bar{\Gamma}\left(n_m - b_m\left(\frac{J(\bar{\Gamma}(n_m + \delta\Delta_m))}{\delta\Delta_m}\right)\right). \tag{14}$$

- The above is analogous to

$$n_{m+1} = \bar{\Gamma}(n_m - b_m(z(n_m) + O(\delta))), \tag{15}$$

where $z(n_m) \in H(n_m)$ with $z(n_m) = \dot{J}(n_m)$ for $n_m \in (k, k+1)$, $k, k+1 \in D$.
- Let $y(\cdot)$ be any bounded perturbed solution to the DI (13).

Main Result (Contd)

- The limit set $L(y) = \cap_{t \geq 0} \overline{\{y(s) | s \geq t\}}$ is then internally chain transitive (cf. Theorem 3.6[14])
- The trajectory obtained from (15) by itself is a bounded and perturbed solution to the DI (13). The claim follows.
- Remark 1: From Theorem 1, if $\hat{\bar{\Gamma}}(-\dot{J}(n^*)) = 0$ for some $n^* \in C \subset \bar{D}$, then $0 \in H(n^*)$ and the recursion (15) will converge to the largest chain transitive invariant set contained in C.
- There are at least two points in D, namely $n = 1$ and $n = L$ for which $0 \in H(n)$. Thus, in general, if the algorithm does not converge to a point in the set $D^o = \{2, 3, \ldots, L-1\}$, it will converge to either $n = 1$ or $n = L$.

[14] M. Benaïm, J. Hofbauer, and S. Sorin, Stochastic approximations and differential inclusions. SIAM Journal on Control and Optimization, 44(1), pp.328-348, 2005.

Numerical Experiments

- Experiments on two RL benchmark environments
  - Random Walk (21 states)[15]
  - Grid World (256 states)[16]
- Multiple experiments run for different initial condition, step-sizes etc.

[15] R.Sutton and A.Barto, Reinforcement Learning, MIT Press, 2018.

[16] M. Chevalier-Boisvert et al., Minigrid & miniworld: Modular & customizable reinforcement learning environments for goal-oriented tasks. NeurIPS, 36, 2024.

Results on Grid World

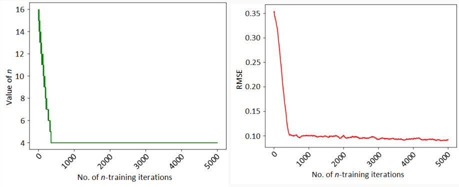


Figure 4: (a) n-updates and (b) RMSE for initial $n = 16$ and $\alpha = 0.4$.

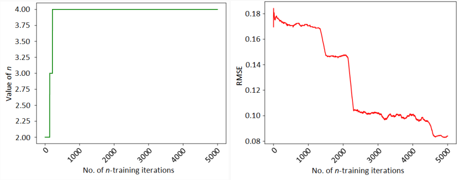


Figure 5: (a) n-updates and (b) RMSE for initial $n = 2$ and $\alpha = 0.4$.

Optimal Computing Budget Allocation[17]

- OCBA is a well known algorithm for discrete parameter stochastic optimization over small and medium sized parameter sets.
- The procedure initially asks for a computing budget and assigns a small budget for initial exploration across parameters.
- Subsequently, over multiple stages it assigns budget based on the current estimates (of value function for different n and it also makes use of RMSE).
- The procedure continues until the computing budget is exhausted.

[17] C.-H. Chen and L. H. Lee, Stochastic Simulation Optimization: An Optimal Computing Budget Allocation. Singapore: World Scientific, 2010

Comparisons in RMSE with OCBA on GW

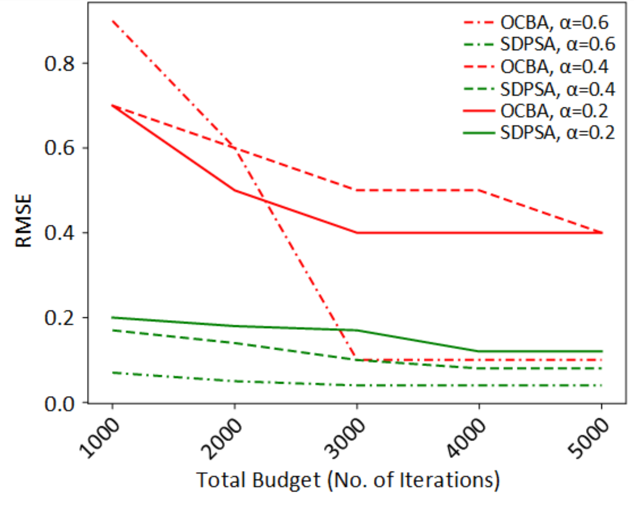


Figure 6: RMSE values w.r.t. computation budget of OCBA and SDPSA with fixed α on GW.

Comparisons with OCBA - RMSE and Computational Time

Table 1: Comparison Results of OCBA and SDPSA on GW.

α in *n*-step TD	Time (Sec.)		RMSE	
	OCBA	SDPSA	OCBA	SDPSA
0.6	588	**452**	0.10	**0.04**
0.4	610	**405**	0.40	**0.08**
0.2	571	**490**	0.40	**0.12**

Conclusions and Future Work

- Devised a two-timescale stochastic approximation scheme to find optimal n in n-step TD learning.
- Gave a proof of convergence.
- Experimental results show better results than OCBA - a well known algorithm for discrete parameter stochastic optimization
- Future work can focus on
    - actor-critic algorithms with n-TD critic with an adaptive n.
    - finding optimal λ for TD(λ) in function approximation schemes.