A Family of Robust Stochastic Operators for Reinforcement Learning

Yingdong Lu, Mark S. Squillante, Chai Wah Wu
Mathematical Sciences
IBM Research
Yorktown Heights, NY 10598, USA
{yingdong, mss, cwwu}@us.ibm.com

Abstract

We consider a new family of stochastic operators for reinforcement learning that seeks to alleviate negative effects and become more robust to approximation or estimation errors. Theoretical results are established, showing that our family of operators preserve optimality and increase the action gap in a stochastic sense. Empirical results illustrate the strong benefits of our robust stochastic operators, significantly outperforming the classical Bellman and recently proposed operators.

1 Introduction

Reinforcement learning has a rich history within the machine learning community to solve a wide variety of decision making problems in environments with unknown and possibly unstructured dynamics. Through iterative application of a convergent operator, value-based reinforcement learning (RL) generates successive refinements of an initial value function [14, 22, 21]. Q-learning [24] is a particular RL technique in which the computations of value iteration consist of evaluating the corresponding Bellman equation without a model of the environment.

While Q-learning continues to be broadly and successfully used in RL to determine the optimal actions of an agent, the development of new Q-learning approaches that improve convergence speed, accuracy and robustness remains of great interest. One area of particular interest concerns environments in which there exist approximation or estimation errors. Of course, when no approximation/estimation errors are present, then the corresponding Markov decision process (MDP) can be solved exactly with the Bellman operator. However, in the presence of nonstationary errors – a typically encountered example being when a discrete-state, discrete-time MDP is used to approximate a continuous-state, continuous-time system – then the optimal state-action value function obtained through the Bellman operator does not always describe the value of stationary policies. Hence, when the optimal state-action value function and the suboptimal state-action value functions are reasonably close to each other, approximation/estimation errors can cause suboptimal actions to be chosen instead of an optimal action and thus in turn potentially causing errors in identifying truly optimal actions.

To help explain and formalize this phenomena, Farahmand [13] introduced the notion of action-gap regularity and showed that a larger action-gap regularity implies a smaller loss in performance. Building on action-gap regularity and its benefits with respect to (w.r.t.) performance loss, Bellemare et al. [6] considered a particular approach to having the value iteration converge to an alternative action-value function Q associated with the same optimal action policy – i.e., maintain properties of optimality-preserving – while at the same time achieving a larger separation between the Q-values of optimal actions and those of suboptimal actions – i.e., maintain properties of action-gap increasing. The former properties ensure optimality whereas the latter properties may assist the value-iteration algorithm to determine the optimal actions of an agent faster, more easily, and with less errors of mislabeling suboptimal actions. Therefore, by exploiting weaker optimality conditions than the

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

Bellman equation and due to the known benefits of larger action-gap regularity, this approach can potentially lead to alternatives to the classical Bellman operator that improve the convergence speed, accuracy and robustness of RL in environments with approximation/estimation errors.

Following this approach, Bellemare et al. [6] propose purely deterministic operator alternatives to the classical Bellman operator and show that the proposed operators satisfy the properties of optimality-preserving and gap-increasing. Then, after empirically demonstrating the benefits of their proposed deterministic operator alternatives, the authors raise a number of open fundamental questions w.r.t. the possibility of weaker conditions for optimality, the statistical efficiency of their proposed operators, and the possibility of finding a maximally efficient operator.

At the heart of the problem is a fundamental tradeoff between the degree to which the preservation of optimality is violated and the degree to which the action gap is increased. Although the benefits of increasing action-gap regularity are known [13], increasing the action gap beyond a certain region in a deterministic sense can lead to violations of optimality preservation (due to deviating too far from Bellman optimality), thus rendering value iterations that may not ensure convergence to optimal solutions. Hence, any purely deterministic operator alternative is unfortunately limited in the degree to which it can be both gap-increasing and optimality-preserving, and thus in turn limited in the degree to which it can address the above problems of approximation/estimation errors in RL.

We therefore consider an approach based on a novel stochastic framework that can increase the action gap well beyond such a deterministic region for individual value iterations – via a random variable (r.v.) – while controlling in a probabilistic manner the overall value iterations – via a sequence of r.v.s – to ensure optimality preservation in a stochastic sense. Our general approach is applicable to arbitrary Q-value approximation schemes in which the sequence of r.v.s provides support to devalue suboptimal actions while preserving the set of optimal policies almost surely (a.s.), thus making it possible to increase the action gap between the Q-values of optimal and suboptimal actions beyond the deterministic region; this can be important in practice because of the potential advantages of increasing the action gap when there are approximation/estimation errors. In devising a family of operators within our framework endowed with these properties, we provide a general stochastic approach that can address the inherent deficiencies of purely deterministic operator alternatives to the classical Bellman operator and that can potentially yield greater robustness w.r.t. mislabeling suboptimal actions in the presence of approximation/estimation errors. To the best of our knowledge, this paper presents the first proposal and theoretical analysis of such types of robust stochastic operators (RSOs), which is an approach not often seen in the RL literature and should be exploited to a much greater extent.

The research literature contains a wide variety of studies of operator alternatives to the Bellman operator, including the ϵ-greedy method [24], speedy Q-learning [3], policy iteration-like Q-learning [8], and the Boltzmann softmax operator and its variants [2]. Each of these operator alternatives seeks to address certain issues in RL. In this paper we complement these previous studies of operator alternatives and focus on operators that seek to achieve greater robustness w.r.t. approximation/estimation errors; in fact, our empirical studies are based on Q-learning with the ϵ-greedy method.

Our theoretical results include proving that our stochastic operators are optimality-preserving and gap-increasing in a stochastic sense. Since the value-iteration sequence generated under our stochastic operators is based on realizations of independent nonnegative r.v.s, our family of RSOs subsumes the family of purely deterministic operators in [6] as a strict subset (because the realizations of r.v.s can be fixed to match that of any deterministic operators as a special case). We further prove that stochastic and variability orderings among the sequence of random operators lead to corresponding orderings among the action gaps. Our stochastic framework and theoretical results shed new light on the open fundamental questions raised in [6], which includes our family of RSOs rendering significantly weaker conditions for optimality and significantly stronger statistical efficiency. Another important implication of our results is that the search space for the maximally efficient operator should be an infinite dimensional space of sequences of r.v.s, instead of the finite dimensional space alluded to in [6]. Yet another important implication is that the order relationships among the sequences of random operators w.r.t. action gaps, corresponding to our stochastic and variability ordering results, may potentially lead to determining the best sequence of r.v.s and possibly even lead to maximally efficient operators. These theoretical results further extend our understanding of the relationship between action-gap regularity and the effectiveness of Q-learning in environments with approximation/estimation errors beyond the initial studies in [13, 6].

We subsequently apply our RSOs to obtain empirical results for various problems in the OpenAI Gym framework [10]. Using these existing codes with minor modifications, we compare the empirical results under our family of stochastic operators against those under both the classical Bellman operator and the consistent Bellman operator [6]. These experiments consistently show that our RSOs outperform both of these deterministic operators. Appendix C of the supplement provides the corresponding python code modifications used in our experiments.

2 Preliminaries

We consider a standard RL framework (see, e.g., [7]) in which a learning agent interacts with a stochastic environment. This interaction is modeled as a discrete-space, discrete-time discounted MDP denoted by $(\mathbb{X}, \mathbb{A}, \mathbb{P}, R, \gamma)$, where $\mathbb{X}$ represents the set of states, $\mathbb{A}$ the set of actions, $\mathbb{P}$ the transition probability kernel, R the reward function mapping state-action pairs into a bounded subset of $\mathbb{R}$, and $\gamma \in [0, 1)$ the discount factor. Let $\mathbb{Q}$ denote the set of bounded real-valued functions over $\mathbb{X} \times \mathbb{A}$. For $Q \in \mathbb{Q}$, define $V(x) := \max_a Q(x, a)$ and use the same definition for variants such as $\hat{Q} \in \mathbb{Q}$ and $\hat{V}(x)$. Let x' always denote the next state r.v. For the current state x in which action a is taken, i.e., $(x, a) \in \mathbb{X} \times \mathbb{A}$, denote by $\mathbb{P}(\cdot|x, a)$ the conditional transition probability for the next state x' and define $\mathbb{E}_{\mathbb{P}} := \mathbb{E}_{x' \sim \mathbb{P}(\cdot|x,a)}$ to be the expectation w.r.t. $\mathbb{P}(\cdot|x, a)$.

A stationary policy $\pi(\cdot|x) : \mathbb{X} \to \mathbb{A}$ defines the distribution of control actions given the current state x, which reduces to a deterministic policy when the conditional distribution renders a constant action for each state x; with slight abuse of notation, we always write the policy $\pi(x)$. The stationary policy π induces a value function $V^\pi : \mathbb{X} \to \mathbb{R}$ and an action-value function $Q^\pi : \mathbb{X} \times \mathbb{A} \to \mathbb{R}$ where $V^\pi(x) := Q^\pi(x, \pi(x))$ defines the expected discounted cumulative reward under policy π starting in state x and where Q^π satisfies the Bellman equation

$$Q^\pi(x, a) = R(x, a) + \gamma \mathbb{E}_{\mathbb{P}} Q^\pi(x', \pi(x')). \tag{1}$$

Our goal is to determine a policy π^* that achieves the optimal value function $V^*(x) := \sup_\pi V^\pi(x), \forall x \in \mathbb{X}$, which also produces the optimal action-value function $Q^*(x, a) := \sup_\pi Q^\pi(x, a), \forall (x, a) \in \mathbb{X} \times \mathbb{A}$. The *Bellman operator* $\mathcal{T}_B : \mathbb{Q} \to \mathbb{Q}$ is defined pointwise as

$$\mathcal{T}_B Q(x, a) := R(x, a) + \gamma \mathbb{E}_{\mathbb{P}} \max_{b \in \mathbb{A}} Q(x', b), \tag{2}$$

or equivalently $\mathcal{T}_B Q(x, a) = R(x, a) + \gamma \mathbb{E}_{\mathbb{P}} V(x')$. The Bellman operator $\mathcal{T}_B$ is known (see, e.g., [7]) to be a contraction mapping in supremum norm, and its unique fixed point coincides with the optimal action-value function, namely $Q^*(x, a) = R(x, a) + \gamma \mathbb{E}_{\mathbb{P}} \max_{b \in \mathbb{A}} Q^*(x', b)$, or equivalently $Q^*(x, a) = R(x, a) + \gamma \mathbb{E}_{\mathbb{P}} V^*(x')$. This in turn indicates that the optimal policy π^* can be obtained by $\pi^*(x) = \arg\max_{a \in \mathbb{A}} Q^*(x, a), \ \forall x \in \mathbb{X}$.

While the Bellman operator can exactly solve the MDP when there are no approximation/estimation errors, the previously noted differences between optimal and suboptimal state-action value functions in the presence of such errors can result in incorrectly identifying the optimal actions. To address these and related nonstationary effects of approximation/estimation errors arising in practice, Bellemare et al. [6] propose the so-called *consistent Bellman operator* defined as

$$\mathcal{T}_C Q(x, a) := R(x, a) + \gamma \mathbb{E}_{\mathbb{P}}[\mathbb{1}_{\{x \neq x'\}} \max_{b \in \mathbb{A}} Q(x', b) + \mathbb{1}_{\{x = x'\}} Q(x, a)], \tag{3}$$

where $\mathbb{1}_{\{\cdot\}}$ denotes the indicator function. The consistent Bellman operator $\mathcal{T}_C$ preserves a local form of stationarity by redefining the action-value function Q such that, if an action $a \in \mathbb{A}$ is taken from the state $x \in \mathbb{X}$ and the next state $x' = x$, then action a is taken again. Bellemare et al. [6] proceed to show that the consistent Bellman operator yields the optimal policy π^*, and in particular that $\mathcal{T}_C$ is both optimality-preserving and gap-increasing, according to (deterministic) definitions that they provide which are compatible with those from Farahmand [13].

The proofs of our theoretical results involve mathematical arguments and technical details that are unique to stochastic operators and stochastic orderings, and distinct from any previous deterministic operators. In particular, a r.v. X is stochastically greater than or equal to ($\geq_{st}$) a r.v. Y if $\mathbb{P}[X > z] \geq \mathbb{P}[Y > z], \forall z$, and a r.v. X is greater than or equal to ($\geq_{cx}$) a r.v. Y under a convex ordering if and only if $\mathbb{E}[f(X)] \geq \mathbb{E}[f(Y)]$, $\forall$ convex functions f. Additional technical details on these and other probabilistic terms and results underlying our theoretical results can be found in [9, 11, 18].

3 Robust Stochastic Operators

In this section we present our stochastic framework which includes proposing a general family of RSOs, providing precise definitions of the concepts of optimality-preserving and gap-increasing in a stochastic sense for a sequence of random operators, and establishing that any sequence of this general family of operators is optimality-preserving and gap-increasing. Our introduction of a new family of operators and our shifting the focus from one deterministic operator to a sequence of stochastic operators has significant implications w.r.t. the open questions raised in [6]. Specifically, our results show that the conditions for optimality are much weaker and the statistical efficiency of our operators can be made much stronger, both allowing significant degrees of freedom in finding alternatives to the Bellman operator for different purposes and applications. Meanwhile, these important improvements completely alter and clarify the question of finding the maximally efficient operators from a finite dimensional parameter optimization problem suggested in [6] to an optimization problem in an infinite dimensional space (of the infinite sequences of r.v.s), for which we establish that stochastic and variability orderings among the sequence of random operators lead to corresponding orderings among the action gaps. It is important to note that our approach can be extended to variants of the Bellman operator such as SARSA [17], policy evaluation [19] and fitted Q-iteration [12].

For all $Q_0 \in \mathbb{Q}$, $x \in \mathbb{X}$, $a \in \mathbb{A}$ and sequences $\{\beta_k : k \in \mathbb{Z}_+\}$ of independent nonnegative r.v.s with expectation $\overline{\beta}_k := \mathbb{E}_\beta[\beta_k]$ between 0 and 1 inclusively for each $k \in \mathbb{Z}+$, we define

$$\mathcal{T}_{\beta_k} Q_k(x,a) := R(x,a) + \gamma \mathbb{E}_{\mathbb{P}} \max_{b \in \mathbb{A}} Q_k(x',b) - \beta_k (V_k(x) - Q_k(x,a)), \tag{4}$$

or equivalently $\mathcal{T}_{\beta_k} Q_k(x,a) := R(x,a) + \gamma \mathbb{E}_{\mathbb{P}} V(x') - \beta_k (V_k(x) - Q_k(x,a))$. (Note that the operator in (4) is equivalent to the Bellman operator whenever the action a is optimal or $\beta_k = 0$, thus making the difference term zero in these cases.) Then members of the general family of RSOs include the sequence $\{\mathcal{T}_{\beta_k}\}$ defined over all probability distributions for each r.v. in the sequence $\{\beta_k\}$ with $\overline{\beta}_k \in [0,1]$. (Note, in particular, that the r.v.s β_k can follow a different probability distribution for each k.) We further define $\mathcal{T}_\beta^{\mathcal{F}}$ to be the general family of RSOs comprising all sequences of operators $\{\mathcal{T}\}$, each as defined in (4), such that there exists a sequence of $\{\beta_k\}$ and, for all $x \in \mathbb{X}$ and $a \in \mathbb{A}$, the following inequalities hold

$$\mathcal{T}_B Q(x,a) - \beta_k (V_k(x) - Q_k(x,a)) \leq \mathcal{T} Q(x,a) \leq \mathcal{T}_B Q(x,a).$$

Observe that, for any (x,a) in (4) where a is not the optimal action, we have $V_k(x) > Q_k(x,a)$ occurring very often (i.e., for many k), causing $Q(x,a)$ to (eventually) deviate more from $V(x)$; otherwise, for a such that $Q(x,a) = V(x)$, then $V_k(x) > Q_k(x,a)$ will only happen relatively rarely, thus not affecting the end value of $V(x)$. Since the value function $V(x)$ does not change but the action-value function $Q(x,a)$ does indeed change, this can lead to a larger action gap and can potentially render more efficient ways of ultimately finding $V(x)$ through the iterative updating of $Q(x,a)$, as indicated in [13, 6]. Moreover, we observe that the multiplier β_k in front of $V_k(x) - Q_k(x,a)$ is desired to be relatively large individually, but its overall efforts should not be so large as to affect the end value of $V(x)$. We therefore introduce a family of RSOs where β_k is allowed to take on any value as long as its average $\overline{\beta}_k$ remains less than or equal to 1. Obviously, these conditions are strictly weaker than those identified in [6] – theirs being purely deterministic and constrained to $[0,1)$, and ours based on r.v.s β_k that can take on values well outside of $[0,1)$. Since the r.v.s β_k need not be identically distributed (with the sole requirement that $\overline{\beta}_k$ is between 0 and 1 inclusively) and since realizations of β_k can take on values far beyond or equal to 1, the family of operators $\mathcal{T}_\beta^{\mathcal{F}}$ clearly subsumes the family of previously identified deterministic operators as a special case.

For the analysis of our family of stochastic operators, we consider the following key definitions.

Definition 3.1. *A sequence of random operators $\{\mathcal{T}_k\}$ for $\mathcal{M} = (\mathbb{X}, \mathbb{A}, \mathbb{P}, R, \gamma)$ is* optimality-preserving in a stochastic sense *if for any $Q_0 \in \mathbb{Q}$ and $x \in \mathbb{X}$, and for the sequence of r.v.s $Q_{k+1} := \mathcal{T}_k Q_k$, the following properties hold: $V_k(x) := \max_{a \in \mathbb{A}} Q_k(x,a)$ converges a.s. to a constant $\hat{V}(x)$ as $k \to \infty$, and for all $a \in \mathbb{A}$, we have a.s.*

$$Q^*(x,a) < V^*(x) \Rightarrow \limsup_{k \to \infty} Q_k(x,a) < \hat{V}(x). \tag{5}$$

Definition 3.2. *A sequence of random operators $\{\mathcal{T}_k\}$ for $\mathcal{M} = (\mathbb{X}, \mathbb{A}, \mathbb{P}, R, \gamma)$ is* gap-increasing in a stochastic sense *if for all $Q_0 \in \mathbb{Q}$, $x \in \mathbb{X}$ and $a \in \mathbb{A}$, the following inequality holds a.s.:*

$$A(x,a) := \liminf_{k \to \infty} [V_k(x) - Q_k(x,a)] \geq V^*(x) - Q^*(x,a). \tag{6}$$

The property of the optimality-preserving definition essentially ensures a.s. that at least one optimal action remains optimal and all suboptimal actions remain suboptimal, while the property of the gap-increasing definition implies robustness when the inequality (6) is strict a.s. for at least one $(x, a) \in \mathbb{X} \times \mathbb{A}$. In particular, as the action gap of an operator increases while remaining optimality-preserving, the end result can be greater robustness to approximation/estimation errors [13].

We next present one of our main theoretical results establishing that our general family of RSOs is both optimality-preserving and gap-increasing in a stochastic sense.

Theorem 3.1. *Let $\mathcal{T}_B$ be the Bellman operator defined in* (2) *and $\{\mathcal{T}_{\beta_k}\}$ a sequence of RSOs as defined in* (4)*. Considering the sequence of r.v.s $Q_{k+1} := \mathcal{T}_{\beta_k} Q_k$ on a sample path basis with $Q_0 \in \mathbb{Q}$, the sequence of operators $\{\mathcal{T}_{\beta_k}\}$ is both optimality-preserving and gap-increasing in a stochastic sense, a.s. Furthermore, all operators in the family $\mathcal{T}_\beta^{\mathcal{F}}$ are optimality-preserving and gap-increasing in a stochastic sense, a.s.*

Even though the stochastic operators in $\mathcal{T}_\beta^{\mathcal{F}}$ are not contraction mappings and therefore do not have a fixed point (as is also true for $\mathcal{T}_C$ [6]), Theorem 3.1 establishes that each of these stochastic operators in $\mathcal{T}_\beta^{\mathcal{F}}$ is still optimality-preserving. Moreover, the definition of $\mathcal{T}_\beta^{\mathcal{F}}$ and Theorem 3.1 significantly enlarge the set of optimality-preserving and gap-increasing operators beyond the purely deterministic operators identified in [6]. In particular, our new sufficient conditions for optimality-preserving operators in a stochastic sense implies that significant deviation from the Bellman operator is possible without loss of optimality; in comparison, the deterministic operator in [6] never allows a value of β_k equal to or greater than 1. More importantly, the definition of $\mathcal{T}_\beta^{\mathcal{F}}$ and Theorem 3.1 imply that the search space for maximally efficient operators is an infinite dimensional space of sequences of r.v.s, instead of the finite dimensional space for maximally efficient operators alluded to in [6]. To this end and due to our stochastic framework, we now establish results on stochastic ordering properties among the sequences of r.v.s $\{\beta_k\}$ that lead to corresponding ordering properties among the action gaps of the random operators. These results offer key relational insights into important orderings of different operators in $\mathcal{T}_\beta^{\mathcal{F}}$, which further demonstrates the benefit of our RSOs and can potentially be exploited in searching for and attempting to find maximally efficient operators in practice.

Theorem 3.2. *For all $\hat{Q}_0 = \tilde{Q}_0 = Q_0 \in \mathbb{Q}$ and for each integer $k \geq 0$, suppose $\hat{Q}_{k+1}$ and $\tilde{Q}_{k+1}$ are respectively updated with two different RSOs $\mathcal{T}_{\hat{\beta}_k}$ and $\mathcal{T}_{\tilde{\beta}_k}$ that are distinguished by $\hat{\beta}_k$ and $\tilde{\beta}_k$ satisfying the stochastic ordering $\hat{\beta}_k \geq_{st} \tilde{\beta}_k$; namely $\hat{Q}_{k+1} = \mathcal{T}_{\hat{\beta}_k} \hat{Q}_k$ and $\tilde{Q}_{k+1} = \mathcal{T}_{\tilde{\beta}_k} \tilde{Q}_k$. Then we have that the action gaps of the two systems are stochastically ordered in the same direction, namely $\hat{A}(x, a) \geq_{st} \tilde{A}(x, a)$.*

Theorem 3.3. *For all $\hat{Q}_0 = \tilde{Q}_0 = Q_0 \in \mathbb{Q}$ and for each integer $k \geq 0$, suppose $\hat{Q}_{k+1}$ and $\tilde{Q}_{k+1}$ are respectively updated with two different RSOs $\mathcal{T}_{\hat{\beta}_k}$ and $\mathcal{T}_{\tilde{\beta}_k}$ that are distinguished by $\hat{\beta}_k$ and $\tilde{\beta}_k$ satisfying the convex ordering $\hat{\beta}_k \geq_{cx} \tilde{\beta}_k$; namely $\hat{Q}_{k+1} = \mathcal{T}_{\hat{\beta}_k} \hat{Q}_k$ and $\tilde{Q}_{k+1} = \mathcal{T}_{\tilde{\beta}_k} \tilde{Q}_k$. Then we have that the action gaps of the two systems are convex ordered in the same direction, namely $\hat{A}(x, a) \geq_{cx} \tilde{A}(x, a)$.*

Theorem 3.4. *For all $\hat{Q}_0 = \tilde{Q}_0 = Q_0 \in \mathbb{Q}$ and for each integer $k \geq 0$, suppose $\hat{Q}_{k+1}$ and $\tilde{Q}_{k+1}$ are respectively updated with two different RSOs $\mathcal{T}_{\hat{\beta}_k}$ and $\mathcal{T}_{\tilde{\beta}_k}$ that are distinguished by $\hat{\beta}_k$ and $\tilde{\beta}_k$ satisfying $\mathbb{E}[\hat{\beta}_k] = \mathbb{E}[\tilde{\beta}_k]$ and $\mathbb{V}ar[\hat{\beta}_k] \leq \mathbb{V}ar[\tilde{\beta}_k]$; namely $\hat{Q}_{k+1} = \mathcal{T}_{\hat{\beta}_k} \hat{Q}_k$ and $\tilde{Q}_{k+1} = \mathcal{T}_{\tilde{\beta}_k} \tilde{Q}_k$. Then we have $\mathbb{V}ar[\hat{Q}_{k+1}] \leq \mathbb{V}ar[\tilde{Q}_{k+1}]$. Furthermore, the action gaps of the two systems have the following properties: $\mathbb{E}[\hat{A}(x, a)] = \mathbb{E}[\tilde{A}(x, a)]$ and $\mathbb{V}ar[\hat{A}(x, a)] \leq \mathbb{V}ar[\tilde{A}(x, a)]$.*

The first two theorems conclude that, among the sequences of β_k that preserve optimality, those stochastically larger and more variable sequences can produce larger action gaps w.r.t. two standard and important stochastic orderings. Theorem 3.4 points out that a larger variance for β_k, with the same fixed mean value, leads to a larger variance for $Q_k(x, a)$ while rendering the same expectation for the action gap and a larger variance in the action gap. We know that, in the limit, the optimal action will maintain its state-action value function. Then, when k is sufficiently large, we can expect that the state-value function $Q_k(x, b^*)$ for the optimal action b^* in state x will be very close to the optimal value $Q^*(x, b^*)$. A larger variance therefore suggests the potential for a greater separation between $Q_k(x, b^*)$ and the state-value function $Q_k(x, a)$ for sub-optimal actions a, and thus the

latter can be understood to have a larger action gap in the limit. Hence, sequences of β_k with large variances can be seen as a very simple instance of the stochastic ordering results.

4 Experimental Results

Within the general RL framework of interest, we consider a standard, yet generic, form for Q-learning so as to cover the various problems empirically examined in this section. Specifically, for all $Q_0 \in \mathbb{Q}$, $x \in \mathbb{X}$, $a \in \mathbb{A}$ and an operator of interest $\mathcal{T}$, we consider the sequence of action-value Q-functions based on the following generic update rule:

$$Q_{k+1}(x,a) = (1-\alpha_k)Q_k(x,a) + \alpha_k \mathcal{T} Q_k(x,a), \tag{7}$$

where α_k is the learning rate for iteration k. Our theoretical results study the behavior of $Q(x,a)$ under a general class of different operators, establishing the benefits of our RSOs over previously proposed operators. We now turn to our empirical comparisons that consist of the Bellman operator $\mathcal{T}_B$, the consistent Bellman operator $\mathcal{T}_C$, and instances of our family of RSOs $\mathcal{T}_{\beta_k}$ under different distributions for the sequence of β_k.

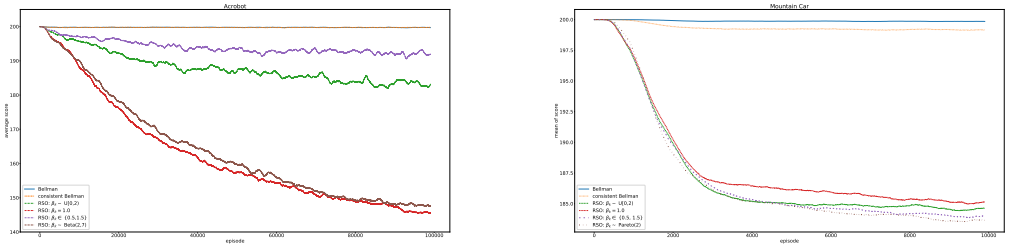
We conduct various experiments across several well-known problems using the OpenAI Gym framework [10], namely Acrobot, Mountain Car, Cart Pole and Lunar Lander. This collection of problems spans a variety of RL examples with different characteristics, dimensions, parameters, and so on. In each case, the state space is continuous and discretized to a finite set of states; i.e., each dimension is discretized into equally spaced bins where the number of bins depends on the problem to be solved and the reference codebase used. For every problem, the specific Q-learning algorithms considered are defined as in (7) where the appropriate operator of interest $\mathcal{T}_B$, $\mathcal{T}_C$ or $\mathcal{T}_{\beta_k}$ is substituted for $\mathcal{T}$; at each timestep, (7) is iteratively applied to the Q-function at the current state and action. The experiments for each problem from the OpenAI Gym were run using the existing code found at [23, 1] exactly as is with the default parameter settings and the *sole* change consisting of the replacement of the Bellman operator in the code with corresponding implementations of either the consistent Bellman operator or RSO; see Appendix C of the supplement for the corresponding python code. It is apparent from these codes that RSO can be directly and easily implemented as a replacement for the classical Bellman operator.

We note that each of the algorithms from the OpenAI Gym implements a form of the ϵ-greedy method (e.g., occasionally picking a random action or using a randomly perturbed Q-function for determining the action) to enable some form of exploration in addition to the exploitation-based search of the optimal policy using the Q-function. Our experiments were therefore repeated over a wide range of values for ϵ, where we found that the relative performance trends of the various operators did not depend significantly on the amount of exploration under the ϵ-greedy algorithm. In particular, the same performance trends were observed over a wide range of ϵ values and hence we present results based on the default value of ϵ used in the reference codebase.

Multiple experimental trials are run for each problem, where we ensured the setting of the random starting state to be the same in each experimental trial for all of the operators considered by initializing them with the same random seed. We observe in general across all experimental results that for different problems and different variants of the Q-learning algorithm, simply replacing the Bellman operator or the consistent Bellman operator with an RSO results in significant performance improvements. The RSOs considered in every set of experimental trials for each problem consist of different distributions for the sequence of β_k. Specifically, we empirically study the following instances of our family of RSOs:

1. β_k sampled from a uniform distribution over $[0,1)$, thus $\mathbb{E}[\beta_k] = \frac{1}{2}$ and $\mathbb{V}\text{ar}[\beta_k] = \frac{1}{12}$;
2. β_k sampled from a uniform distribution over $[0,2)$, thus $\mathbb{E}[\beta_k] = 1$ and $\mathbb{V}\text{ar}[\beta_k] = \frac{1}{3}$;
3. β_k sampled from a uniform distribution over $[0.5,1.5)$, thus $\mathbb{E}[\beta_k] = 1$ and $\mathbb{V}\text{ar}[\beta_k] = \frac{1}{12}$;
4. β_k set to $\frac{3}{5}$ plus a r.v. sampled from a Beta$(2,3)$ distribution, thus $\mathbb{E}[\beta_k] = 1$ and $\mathbb{V}\text{ar}[\beta_k] = \frac{1}{25}$;
5. β_k set to $\frac{7}{9}$ plus a r.v. sampled from a Beta$(2,7)$ distribution, thus $\mathbb{E}[\beta_k] = 1$ and $\mathbb{V}\text{ar}[\beta_k] = \frac{7}{405}$;
6. β_k set to a r.v. sampled from a Pareto$(1,2)$ distribution minus 1, thus $\mathbb{E}[\beta_k] = 1$, $\mathbb{V}\text{ar}[\beta_k] = \infty$;
7. β_k set to a r.v. sampled from a Pareto$(1,3)$ distribution minus $\frac{1}{2}$, thus $\mathbb{E}[\beta_k] = 1$, $\mathbb{V}\text{ar}[\beta_k] = \frac{3}{4}$;
8. β_k set to 0.5 and 1.5 in an alternating manner, thus having $\mathbb{E}[\beta_k] = 1$ and $\mathbb{V}\text{ar}[\beta_k] = \frac{1}{12}$;
9. β_k set to 1, thus having $\mathbb{E}[\beta_k] = 1$ and $\mathbb{V}\text{ar}[\beta_k] = 0$.

Observe that the first and second RSO instances include values of β_k that are equal or relatively close to 0; since $x_m = 1$ in the sixth instance together with the subtraction of 1, this also includes values of β_k that are equal or relatively close to 0; all other RSO instances exclude values of β_k that are equal or relatively close to 0. We note that the last RSO instance is consistent with the advantage learning operator in [4, 6], though it is important to note that $\beta = 1$ was disallowed in [6], unnecessarily so as our results have established. To gain insight on the different RSO instances, the results presented in this section focus on the simple case of operators $\mathcal{T}_{\beta_k}$ associated with sequences of r.v.s $\{\beta_k\}$ drawn from specific probability distributions in an independent and identically distributed manner. We note, however, that various experiments were performed with very simple combinations of different distributions for β_k over the iterations $k \in \mathbb{Z}_+$. As a specific example, we considered $\beta_k \sim U[0, 1)$ for $\beta_0, \ldots, \beta_{k'}$ and then $\beta_k \sim U[0, 2)$ for $\beta_{k'+1}, \ldots$, but these results were not considerably better, and often worse, than those presented below for $\beta_k \sim U[0, 2)$.



(a) Acrobot problem (training).

(b) Mountain Car problem (training).

Figure 1: Average number of steps needed to solve minimization problems during training phase.

4.1 Acrobot

This problem is first discussed in [20]. The state vector is 6-dimensional with three actions possible in each state, and the score represents the number of timesteps needed to solve the problem. The position and velocity are discretized into 8 bins whereas the other state components are discretized into 10 bins. We ran 50 experimental trials over many episodes, with a goal of *minimizing* the score.

Figure 1a plots the score, averaged over moving windows of 1000 episodes across the 50 trials, as a function of the number of episodes for a subset of operators during the training phase; the full set of results are provided in Figure 3. We observe that the average score under the RSOs generally exhibit much better performance than under the Bellman operator or the consistent Bellman operator, with the β_k sequences of all ones and from Beta$(2, 7)$ rendering the best performance. Table 1 presents the average score over the last 1000 episodes across the 50 trials together with the corresponding 95% confidence intervals. We observe that the confidence intervals for all operators are quite small and that the best average scores are consistent with those plotted in Figure 3.

Figure 2b presents the average score over 1000 episodes across the 50 trials for all operators during the testing phase, together with the corresponding 95% confidence intervals. We again observe that the best average scores are obtained under many of the RSOs and that the confidence intervals for all operators are quite small. We further observe the differences in the performance orderings among the operators in comparison with the results in Table 1, where the β_k sequences from Pareto$(1, 2)$ and alternating 0.5 and 1.5 render the best performance followed by β_k sequences from $U[0, 1)$.

4.2 Mountain Car

This problem is first discussed in [16]. The state vector is 2-dimensional with a total of three possible actions, and the score represents the number of timesteps needed to solve the problem. The state space is discretized into a 40×40 grid. We ran 50 experimental trials over many episodes for training, each of which consists of up to 200 steps and with a goal of *minimizing* the score.

Figure 1b plots the score, averaged over moving windows of 1000 episodes across the 50 trials, as a function of the number of episodes for a subset of operators during the training phase; the full set

of results are provided in Figure 4. We observe that the average score under the RSOs generally exhibit considerably better performance than under the Bellman operator or the consistent Bellman operator, with the β_k sequences from Pareto$(1, 2)$ and alternating 0.5 and 1.5 rendering the best performance followed by β_k sequences from $U[0, 2)$. Table 1 presents the average score over the last 1000 episodes across the 50 trials together with the corresponding 95% confidence intervals. We observe that the confidence intervals for all operators are quite small and that the best average scores are consistent with those plotted in Figure 4.

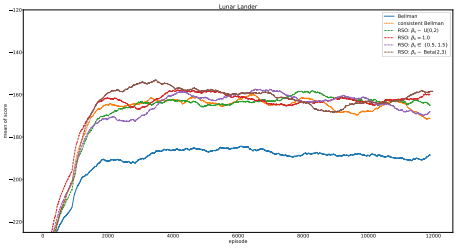
Figure 2b presents the average score over 1000 episodes across the 50 trials for all operators during the testing phase, together with the corresponding 95% confidence intervals. We again observe that the best average scores are generally obtained under the RSOs and that the confidence intervals for all operators are quite small. We further observe the differences in the average score performance orderings among the operators in comparison with the results in Table 1, where the β_k sequences from Pareto$(1, 3)$ and $U[0, 2)$ render the best average score performance.

4.3 Cart Pole

This problem is first discussed in [5]. The state vector is 4-dimensional with two actions possible in each state, and the score represents the number of steps where the cart pole stays upright before either falling over or going out of bounds. The position and velocity are discretized into 8 bins whereas the angle and angular velocity are discretized into 10 bins. We ran 50 experimental trials over many episodes, each of which consists of up to 200 steps with a goal of *maximizing* the score. The problem is considered solved when the score exceeds 195.

Table 1 presents the average score over the last 1000 episodes across the 50 trials for all operators during the training phase, together with the corresponding 95% confidence intervals. We observe that the best average scores are obtained under many of the RSOs, with the β_k sequences of all ones and from Beta$(2, 7)$ rendering the best performance followed by β_k sequences from $U[0.5, 1.5)$. We further observe that the confidence intervals for all operators are quite small.

Table 2 presents the average score over 1000 episodes across the 50 trials for all operators during the testing phase, together with the corresponding 95% confidence intervals. We again observe that the best average scores are obtained under many of the RSOs and that the confidence intervals for all operators are quite small. We further observe the differences in the average score performance orderings among the operators in comparison with the results in Table 1, where the β_k sequences from $U[0.5, 1.5)$ and $U[0, 2)$ render the best average score performance.



(a) Average Lunar Lander score (training).

Testing Score	Acrobot	Mountain Car	Lunar Lander
Bellman	$189.1 \pm 0.17\%$	$131.2 \pm 0.23\%$	$-231.0 \pm 0.92\%$
Consistent Bellman	$185.3 \pm 0.20\%$	$127.2 \pm 0.22\%$	$-185.1 \pm 0.98\%$
$\beta_k \sim U[0, 2)$	$189.5 \pm 0.16\%$	*121.2* $\pm 0.21\%$	$-164.4 \pm 1.05\%$
$\beta_k \sim U[0, 1)$	$184.9 \pm 0.18\%$	$126.9 \pm 0.23\%$	$-207.0 \pm 0.94\%$
$\beta_k = 1.0$	$189.2 \pm 0.18\%$	$121.9 \pm 0.21\%$	$\mathbf{-157.8} \pm 1.10\%$
$\beta_k \in \{0.5, 1.5\}$	*181.3* $\pm 0.23\%$	$122.3 \pm 0.20\%$	$-174.0 \pm 1.01\%$
$\beta_k \sim U[0.5, 1.5)$	$192.4 \pm 0.13\%$	$122.8 \pm 0.21\%$	$-168.1 \pm 1.08\%$
$\beta_k \sim$ Beta(2,3)	$185.0 \pm 0.20\%$	$122.6 \pm 0.21\%$	*-163.5* $\pm 1.13\%$
$\beta_k \sim$ Beta(2,7)	$186.2 \pm 0.19\%$	$122.3 \pm 0.21\%$	$-164.8 \pm 1.06\%$
$\beta_k \sim$ Pareto(2)	$\mathbf{180.7} \pm 0.37\%$	$125.0 \pm 0.20\%$	$-216.9 \pm 0.94\%$
$\beta_k \sim$ Pareto(3)	$186.6 \pm 0.21\%$	$\mathbf{121.1} \pm 0.21\%$	$-166.2 \pm 1.04\%$

(b) Table of average scores (testing).

Figure 2: Average number of steps needed to solve Lunar Lander maximization problem during training phase; Average scores for all RSO instances and three problems during testing phase.

4.4 Lunar Lander

This problem is discussed in [10]. The state vector is 8-dimensional with a total of four possible actions, and the physics of the problem is known to be notoriously more difficult than the foregoing problems. The 6 continuous state variables are each discretized into 4 bins. The score represents the cumulative reward comprising positive points for successful degrees of landing and negative points for fuel usage and crashing. We ran 50 experimental trials over many episodes, each of which consists of up to 200 steps with a goal of *maximizing* the score.

Figure 2a plots the score, averaged over moving windows of 1000 episodes across the 50 trials, as a function of the number of episodes for a subset of operators during the training phase; the full set of results are provided in Figure 5. We observe that the average score under the RSOs generally exhibit better performance than under the Bellman operator or the consistent Bellman operator, with the β_k sequences from Beta$(2, 3)$ and of all ones rendering the best performance. Table 1 presents the average score over the last 1000 episodes across the 50 trials together with the corresponding 95% confidence intervals. We observe that the confidence intervals for all operators are quite small and that the best average scores are consistent with those plotted in Figure 5.

Figure 2b presents the average score over 1000 episodes across the 50 trials for all operators during the testing phase, together with the corresponding 95% confidence intervals. We again observe that the best average scores are generally obtained under the RSOs and that the confidence intervals for all operators are quite small. We further observe some consistencies in the performance orderings among the operators in comparison with the results in Table 1, where the β_k sequences of all ones and from Beta$(2, 3)$ render the best performance followed by β_k sequences from $U[0, 2)$.

5 Conclusions and Discussion

Building on the work of Farahmand [13] and Bellemare et al. [6], who argue that increasing the action gap while preserving optimality can improve the performance of value-iteration algorithms in environments with approximation or estimation errors, we propose and analyze a new general family of RSOs for RL that subsumes as a strict subset the classical Bellman operator and other purely deterministic operators proposed in the literature. Our theoretical results include proving that our stochastic operators are optimality-preserving and gap-increasing in a stochastic sense and that stochastic and variability orderings among the sequence of random operators lead to corresponding orderings among the action gaps. In addition, our stochastic framework and theoretical results shed new light on and help to resolve the open fundamental questions raised in [6] related to the possibility of weaker optimality conditions, the statistical efficiency of proposed deterministic operators, and the possibility of finding maximally efficient operators. Specifically, our theoretical results show that the conditions for optimality are much weaker and the statistical efficiency of our stochastic operators can be made much stronger, both allowing significant degrees of freedom in finding alternatives to the Bellman operator for different purposes and applications. Meanwhile, these important improvements completely alter and clarify the question of finding the maximally efficient operators from a finite dimensional parameter optimization problem suggested in [6] to an optimization problem in an infinite dimensional space (of the infinite sequences of r.v.s), for which our established stochastic and variability orderings among sequences of random operators can potentially assist in searching for maximally efficient operators in practice. Our family of RSOs represents a stochastic approach not often seen in the RL literature that should be exploited to a much greater extent.

A collection of empirical results – based on well-known problems within the OpenAI Gym framework spanning various RL examples with diverse characteristics – support our theoretical results, consistently demonstrating and quantifying the significant performance improvements obtained with our RSOs over existing operators. We note that, while the focus of our empirical results has been on Q-learning, our family of RSOs are applicable to other RL approaches such as DQN [15].

It is important to highlight a few fundamental tradeoffs in identifying maximally efficient operators for different RL problems, based on our theoretical and empirical results. On the one hand, when sampled values of β_k are relatively small, then it is possible for the small offset by $\beta_k(V_k(x) - Q_k(x, a))$ on truly suboptimal actions a to have limited or no effect on the separation between optimal and suboptimal actions. On the other hand, when sampled values of β_k are relatively large, then it is possible for the large offset of $\beta_k(V_k(x) - Q_k(x, a))$ to be applied against the truly optimal action a^* due to approximation or estimation errors. In addition, the level of impact of these and related factors associated with the sequence of r.v.s $\{\beta_k\}$ can vary over the value iterations moving from $k = 0$ to the limit as $k \to \infty$. We view the problem of finding maximally efficient operators for RL problems as identifying sequences of random operators that address these fundamental tradeoffs in order to maximize action-gap regularity for the suboptimal actions of each state. Our theoretical and empirical results further raise a related fundamental issue that concerns whether maximizing the action gap is sufficient to improve the performance of value-iteration algorithms in environments with approximation or estimation errors.

References

[1] M. Alzantot. Solution of mountaincar OpenAI Gym problem using Q-learning. *https://gist.github.com/malzantot /9d1d3fa4fdc4a101bc48a135d8f9a289*, 2017.

[2] K. Asadi and M. L. Littman. An alternative softmax operator for reinforcement learning. In *Proc. 34th International Conference on Machine Learning*, 2017.

[3] M. Azar, R. Munos, M. Gavamzadeh, and H. Kappen. Speedy Q-learning. *Advances in Neural Information Processing Systems*, 24, 2011.

[4] L. Baird. *Reinforcement Learning through Gradient Descent*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA, U.S.A., 1999.

[5] A. G. Barto, R. S. Sutton, and C. W. Anderson. Neuronlike adaptive elements that can solve difficult learning control problems. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-13(5):834–846, Sept. 1983.

[6] M. G. Bellemare, G. Ostrovski, A. Guez, P. S. Thomas, and R. Munos. Increasing the action gap: New operators for reinforcement learning. In *Proc. Thirtieth AAAI Conference on Artificial Intelligence*, AAAI'16, pages 1476–1483. AAAI Press, 2016.

[7] D. Bertsekas and J. Tsitsiklis. *Neuro-Dynamic Programming*. Athena Scientific, 1996.

[8] D. Bertsekas and H. Yu. Q-learning and enhanced policy iteration in discounted dynamic programming. *Mathematics of Operations Research*, 37, 2012.

[9] P. Billingsley. *Convergence of Probability Measures*. Wiley, New York, Second edition, 1999.

[10] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba. OpenAI Gym. *CoRR*, abs/1606.01540, 2016.

[11] Y. S. Chow and H. Teicher. *Probability Theory: Independence, Interchangeability, Martingales*. Springer, 3rd edition, 2003.

[12] D. Ernst, P. Geurts, and L. Wehenkel. Tree-based batch mode reinforcement learning. *Journal of Machine Learning Research*, 6:503–556, 2005.

[13] A. Farahmand. Action-gap phenomenon in reinforcement learning. *Advances in Neural Information Processing Systems*, 24, 2011.

[14] L. Kaelbling, M. Littman, and A. Moore. Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, 4:237–285, 1996.

[15] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller. Playing Atari with deep reinforcement learning. In *Proc. NIPS Deep Learning Workshop*, 2013.

[16] A. Moore. *Efficient Memory-Based Learning for Robot Control*. PhD thesis, University of Cambridge, Cambridge, U.K., 1990.

[17] G. Rummery and M. Niranjan. On-line Q-learning using connectionist systems. Technical report, Cambridge University, 1994.

[18] M. Shaked and J. Shanthikumar. *Stochastic Orders*. Springer Series in Statistics. Springer New York, 2007.

[19] R. Sutton. Learning to predict by the methods of temporal differences. *Machine Learning*, 3(1):9–44, 1988.

[20] R. S. Sutton. Generalization in Reinforcement Learning: Successful Examples Using Sparse Coarse Coding. *Advances in Neural Information Processing Systems*, 8:1038–1044, 1996.

[21] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 2011.

[22] C. Szepesvari. Algorithms for reinforcement learning. In *Synthesis Lectures on Artificial Intelligence and Machine Learning*, volume 4.1, pages 1–103. Morgan & Claypool, 2010.

[23] V. M. Vilches. Basic reinforcement learning tutorial 4: Q-learning in OpenAI Gym. *https://github.com/vmayoral /basic_reinforcement_learning/blob/master/tutorial4 /README.md*, May 2016.

[24] C. Watkins. *Learning from Delayed Rewards*. PhD thesis, University of Cambridge, Cambridge, U.K., 1989.

A Proofs of Theoretical Results

In this section we present the proofs of our main theoretical results.

A.1 Proof of Theorem 3.1

For any $x \in \mathbb{X}$, define $\pi^{b_k}(x) := \arg\max_b Q_k(x,b)$ on each sample path ω of the stochastic operator $\{\mathcal{T}_{\beta_k}\}$. By the definition of $Q_k(x,a)$, and since $V_k(x) \geq Q_k(x,a)$ for all a by the definition of $V_k(x)$, we have that $\mathcal{T}_{\beta_k} Q_k(x,a) \leq \mathcal{T}_B Q_k(x,a)$ and $\mathcal{T}_{\beta_k} Q_k(x,\pi^{b_k}(x)) = \mathcal{T}_B Q_k(x,\pi^{b_k}(x))$, both a.s. Although π^{b_k} depends on the state x, we will omit the argument x in what follows for ease of exposition.

Making use of the facts that $Q_k(x,\pi^{b_k}) = \mathcal{T}_{\beta_k} Q_{k-1}(x,\pi^{b_k}) = V_k(x)$, we then derive

$$\begin{aligned} V_{k+1}(x) - V_k(x) &\geq Q_{k+1}(x,\pi^{b_k}) - Q_k(x,\pi^{b_k}) \\ &= \mathcal{T}_{\beta_k} Q_k(x,\pi^{b_k}) - \mathcal{T}_{\beta_k} Q_{k-1}(x,\pi^{b_k}) = \mathcal{T}_B Q_k(x,\pi^{b_k}) - \mathcal{T}_{\beta_k} Q_{k-1}(x,\pi^{b_k}) \\ &= \mathcal{T}_B Q_{k-1}(x,\pi^{b_k}) + \gamma \mathbb{E}_{\mathbb{P}}[V_k(x') - V_{k-1}(x')|x,\pi^{b_k}] - \mathcal{T}_{\beta_k} Q_{k-1}(x,\pi^{b_k}) \\ &\geq \gamma \mathbb{E}_{\mathbb{P}}[V_k(x') - V_{k-1}(x')|x,\pi^{b_k}], \end{aligned}$$

where the third line follows directly from the definition of $\mathcal{T}_B$ and the fourth line is directly due to the order relation of $\mathcal{T}_B$ and $\mathcal{T}_{\beta_k}$. This renders

$$V_{k+1}(x) - V_k(x) \geq \gamma \mathbb{E}_{\mathbb{P}}[V_k(x') - V_{k-1}(x')|x,\pi^{b_k}],$$

and by induction we obtain

$$V_{k+1}(x) - V_k(x) \geq \gamma^k \mathbb{E}_{\mathbb{P}}[V_1(x') - V_0(x')|x,\pi^{b_1},\ldots,\pi^{b_k}],$$

from which we conclude

$$V_{k+1}(x) - V_k(x) \geq -\gamma^k ||V_1(x') - V_0(x')||_\infty. \tag{8}$$

Define $f_k = ||V_1(x') - V_0(x')||_\infty \sum_{\ell=0}^{k-1} \gamma^\ell$ for $k \in \mathbb{Z}^+$. Obviously, $V_k(x) + f_k$ is uniformly upper bounded from the facts that the rewards are bounded functions and $\gamma \in (0,1)$. Then (8) implies that $V_k(x) + f_k$ is monotone, and thus it will converge. Meanwhile, f_k obviously converges to $||V_1(x') - V_0(x')||_\infty/(1-\gamma)$, which leads to the a.s. convergence of $V_k(x)$.

Given the a.s. convergence of $V_k(x)$, we now need to identify its limit. The probabilistic nature of the stochastic operators makes it possible to leverage different forms of convergence of measures for the corresponding sequence of r.v.s. Specifically, this probabilistic nature affords us the liberty to exploit weak convergence limits (convergence in probability) to identify the limit of $V_k(x)$ after establishing above the stronger a.s. convergence for $V_k(x)$, since a.s. convergence naturally implies that $V_k(x)$ also weakly converges to the limit. Namely, it suffices for us to establish the limit of $V_k(x)$ under convergence in probability which, although a weaker form of convergence, leads to the same limit as that for a.s. convergence. We therefore need to show that, for any $\epsilon > 0$, $\lim_{k\to\infty} \mathbb{P}[|V_k(x) - V^*(x)| > \epsilon] = 0$. Denoting $\hat{V}(x) = \lim_{k\to\infty} V_k(x)$ and defining

$$\hat{Q}(x,a) := \limsup_{k\to\infty} Q_k(x,a) = \limsup_{k\to\infty} \mathcal{T}_{\beta_k} Q_k(x,a),$$

it is readily apparent that we simply need to show

$$\mathbb{P}[\hat{Q}(x,a) - \mathcal{T}_B \hat{Q}(x,a) > \epsilon] = 0 \quad \text{and} \quad \mathbb{P}[\hat{Q}(x,a) - \mathcal{T}_B \hat{Q}(x,a) < -\epsilon] = 0, \tag{9}$$

since (9) leads to $\hat{V}(x) = \max_a\{R(x,a)\} + \gamma \mathbb{E}[\hat{V}(x')]$, which is the equation that is uniquely satisfied by $V^*(x)$.

Let us show the first part of (9), and the second part can be argued similarly. Observe the statement of $\mathbb{P}[\hat{Q}(x,a) - \mathcal{T}_B \hat{Q}(x,a) > \epsilon] = 0$ is actually equivalent to

$$\{Q_k(x,a) \leq \mathcal{T}_B Q_k(x,a) + \epsilon\}$$

happens infinitely often as k goes to infinity. The later is true due to the fact that

$$\{Q_{k+1}(x,a) \leq \mathcal{T}_B Q_k(x,a) + \frac{\epsilon}{2}\} \cup \{|Q_{k+1}(x,a) - Q_k(x,a)| \leq \frac{\epsilon}{2}\} \subseteq \{Q_k(x,a) \leq \mathcal{T}_B Q_k(x,a) + \epsilon\}$$

and the fact that both $\{Q_{k+1}(x,a) \leq \mathcal{T}_B Q_k(x,a) + \frac{\epsilon}{2}\}$ and $\{|Q_{k+1}(x,a) - Q_k(x,a)| \leq \frac{\epsilon}{2}\}$ happen infinitely often. The first one is due to the definition of $Q_k(x,a)$ and the second one is due to the convergence of the subsequence corresponding to the limit superior.

Hence, the desired relationship $\hat{V}(x) = V^*(x)$ holds a.s., which establishes the preservation of optimality.

Now, turning to prove that $\mathcal{T}_{\beta_k}$ is gap-increasing in a stochastic sense, the above arguments render $\lim_{k\to\infty} V_k(x) = V^*(x)$ a.s., and thus (6) is equivalent to $\limsup_{k\to\infty} Q_k(x,a) \leq Q^*(x,a)$ a.s. This inequality follows on a sample path basis from $\mathcal{T}_{\beta_k} Q(x,a) \leq \mathcal{T}_B Q(x,a)$ by definition and our above arguments, and thus we have the desired result for the operators $\mathcal{T}_{\beta_k}$. Furthermore, it is readily verified that the above arguments can be similarly applied to cover all of the operators in $\mathcal{T}_\beta^{\mathcal{F}}$.

Lastly, from the above results of (6) and $\hat{V}(x) = V^*(x)$ a.s., it follows that (5) also holds a.s. for $\mathcal{T}_{\beta_k}$ as well as for all operators in $\mathcal{T}_\beta^{\mathcal{F}}$, thus completing the proof.

A.2 Proof of Theorem 3.2

First, we prove that $\hat{Q}_k \leq_{st} \tilde{Q}_k$ holds for every $k \geq 0$, arguing by induction where the relationship obviously holds for $k = 0$. Suppose that this holds true for certain k, then the identity

$$Q_{k+1}(x,a) = R(x,a) + \gamma \mathbb{E}_{\mathbb{P}} \max_{b\in\mathbb{A}} Q_k(x',b) - \beta_k(V_k(x) - Q_k(x,a)),$$

together with the fact that $\hat{\beta}_k \geq_{st} \tilde{\beta}_k$, yields

$$\mathbb{E}[f(\hat{Q}_{k+1})|\hat{Q}_k] \leq \mathbb{E}[f(\tilde{Q}_{k+1})|\tilde{Q}_k]$$

for any increasing function $f(\cdot)$, as long as the expectations exist. Furthermore, by the induction assumption, we can conclude that $\mathbb{E}[f(\hat{Q}_{k+1})] \leq \mathbb{E}[f(\tilde{Q}_{k+1})]$, and therefore $\hat{Q}_k \leq_{st} \tilde{Q}_k$ because of the properties of $f(\cdot)$ and the definition of stochastic ordering ($\geq_{st}$). Meanwhile, for any state-action pair (x,a) in these systems, the action gap is characterized by the quantity

$$\liminf_{k\to\infty} V_k(x) - Q_k(x,a).$$

Equivalently, we have

$$V^*(x) - \limsup_{k\to\infty} Q_k(x,a),$$

since we know that for both sequences $\{\hat{\beta}_k\}$ and $\{\tilde{\beta}_k\}$, the RSO is optimality preserving. We therefore obtain

$$\mathbb{E}[f(V^*(x) - \limsup_{k\to\infty} \hat{Q}_{k+1})] \geq \mathbb{E}[f(V^*(x) - \limsup_{k\to\infty} \tilde{Q}_{k+1})]$$

for any increasing function, which follows from the fact that the limit preserves the stochastic order. Hence, the stochastic order of the action gap is established.

A.3 Proof of Theorem 3.3

We follow along similar lines for the proof of Theorem 3.2, but for convex ordering. With $f(x)$ being a convex function (and so is $f(-x)$) and with the identity

$$Q_{k+1}(x,a) = R(x,a) + \gamma \mathbb{E}_{\mathbb{P}} \max_{b\in\mathbb{A}} Q_k(x',b) - \beta_k(V_k(x) - Q_k(x,a)),$$

we can prove by induction that $\hat{Q}_k \geq_{cx} \tilde{Q}_k$. Then, for any convex function $f(\cdot)$, we have

$$\mathbb{E}[f(V^*(x) - \limsup_{k\to\infty} \hat{Q}_{k+1})] \geq \mathbb{E}[f(V^*(x) - \limsup_{k\to\infty} \tilde{Q}_{k+1})],$$

and thus establishing the convex order of the action gaps.

A.4 Proof of Theorem 3.4

Let us start by showing that $\mathbb{V}\text{ar}[\hat{Q}_k] \leq \mathbb{V}\text{ar}[\tilde{Q}_k]$, for any $k \geq 0$. Proceeding by induction where the result trivially holds when $k = 0$, we assume the result holds for any k and we examine $\mathbb{V}\text{ar}[\hat{Q}_k]$ and $\mathbb{V}\text{ar}[\tilde{Q}_k]$. We can readily see that

$$\begin{aligned}\mathbb{V}\text{ar}[\hat{Q}_{k+1}] &= \mathbb{E}[\mathbb{V}\text{ar}[\hat{Q}_{k+1}|\hat{Q}_k] + \mathbb{V}\text{ar}[\mathbb{E}[\hat{Q}_{k+1}|\hat{Q}_k]] \\ &= \mathbb{V}\text{ar}[\hat{\beta}_k]\mathbb{E}[(\hat{V}_k(x) - \hat{Q}_k(x,a))^2] + \mathbb{V}\text{ar}[\mathbb{E}[\hat{Q}_{k+1}|\hat{Q}_k]]\end{aligned}$$

and

$$\begin{aligned}\mathbb{V}\text{ar}[\tilde{Q}_{k+1}] &= \mathbb{E}[\mathbb{V}\text{ar}[\tilde{Q}_{k+1}|\tilde{Q}_k] + \mathbb{V}\text{ar}[\mathbb{E}[\tilde{Q}_{k+1}|\tilde{Q}_k]] \\ &= \mathbb{V}\text{ar}[\tilde{\beta}_k]\mathbb{E}[(\tilde{V}_k(x) - \tilde{Q}_k(x,a))^2] + \mathbb{V}\text{ar}[\mathbb{E}[\tilde{Q}_{k+1}|\tilde{Q}_k]],\end{aligned}$$

and therefore $\mathbb{V}\text{ar}[\hat{Q}_{k+1}] \leq \mathbb{V}\text{ar}[\tilde{Q}_{k+1}]$.

Next, for any $\epsilon > 0$, we know that $|\mathbb{E}[\hat{V}_k(x)] - \mathbb{E}[\tilde{V}_k(x)]| < \epsilon$ for sufficiently large k and for any x, due to the a.s. convergence of both $\hat{V}_k(x)$ and $\tilde{V}_k(x)$ to $V^*(x)$ together with their uniform boundedness. We then can conclude, for sufficiently large k, that $|\mathbb{E}[\hat{Q}_k(x,a)] - \mathbb{E}[\tilde{Q}_k(x,a)]| < \epsilon$ from the expressions

$$\begin{aligned}\mathbb{E}[\hat{Q}_k(x,a)] &= [\mathbb{E}[\mathcal{T}_B\hat{Q}_k|\hat{Q}_k]] + \mathbb{E}[\hat{\beta}_k]\mathbb{E}[\mathbb{E}[\hat{V}_k(x) - \hat{Q}_k(x,a)|\hat{Q}_k]], \\ \mathbb{E}[\tilde{Q}_k(x,a)] &= [\mathbb{E}[\mathcal{T}_B\tilde{Q}_k|\tilde{Q}_k]] + \mathbb{E}[\tilde{\beta}_k]\mathbb{E}[\mathbb{E}[\tilde{V}_k(x) - \tilde{Q}_k(x,a)|\tilde{Q}_k]].\end{aligned}$$

Now, recalling the definition in (6) that $A(x,a) := \liminf_{k\to\infty}[V_k(x) - Q_k(x,a)]$ for any (x,a), and considering any converging subsequence of $\hat{Q}_{n_k}(x,a)$, we have

$$\begin{aligned}\mathbb{E}[\lim_{n_k\to\infty}[V_{n_k}(x) - \hat{Q}_{n_k}(x,a)]] &= \lim_{n_k\to\infty}\mathbb{E}[V_{n_k}(x) - \hat{Q}_{n_k}(x,a)] \\ &\geq \lim_{n_k\to\infty}\mathbb{E}[V_{n_k}(x) - \tilde{Q}_{n_k}(x,a)] - \epsilon \\ &\geq \mathbb{E}[\liminf_{n_k\to\infty}[V_{n_k}(x) - \tilde{Q}_{n_k}(x,a)]] - \epsilon \geq \mathbb{E}[\tilde{A}(x,a)] - \epsilon,\end{aligned}$$

where the interchange of limit and expectation in the first equality is due the uniform boundedness of $\hat{Q}_k(x,a)$, the first inequality is due to the above relationship between $\mathbb{E}[\hat{V}_k(x)]$ and $\mathbb{E}[\tilde{V}_k(x)]$ as well as $\mathbb{E}[\hat{Q}_k(x,a)]$ and $\mathbb{E}[\tilde{Q}_k(x,a)]$, the second inequality is due to Fatou's Lemma, and the last inequality is a basic property of limit inferior. Since this is true for any converging subsequence, it holds for the subsequence that achieves $\hat{A}(x,a)$, and therefore we have $\mathbb{E}[\hat{A}(x,a)] \geq \mathbb{E}[\tilde{A}(x,a)] - \epsilon$. Meanwhile, we can apply the exact same arguments in a similar manner to also conclude that $\mathbb{E}[\tilde{A}(x,a)] \geq \mathbb{E}[\hat{A}(x,a)] - \epsilon$. We therefore have $\mathbb{E}[\hat{A}(x,a)] = \mathbb{E}[\tilde{A}(x,a)]$ since ϵ is arbitrary.

The desired result on the variance ordering will follow by showing that $\mathbb{E}[\hat{A}(x,a)^2] \leq \mathbb{E}[\tilde{A}(x,a)^2]$. For this purpose, again consider any converging subsequence of $\tilde{Q}_{n_k}(x,a)$. We then similarly have

$$\begin{aligned}\mathbb{E}\left[\lim_{n_k\to\infty}(V_{n_k}(x) - \tilde{Q}_{n_k}(x,a))^2\right] &= \lim_{n_k\to\infty}\mathbb{E}\left[(V_{n_k}(x) - \tilde{Q}_{n_k}(x,a))^2\right] \\ &= \lim_{n_k\to\infty}\mathbb{E}[V^*(x) - \tilde{Q}_{n_k}(x,a)]^2 \\ &\geq \lim_{n_k\to\infty}\mathbb{E}[V^*(x) - \hat{Q}_{n_k}(x,a)]^2 \\ &\geq \mathbb{E}\left[\liminf_{n_k\to\infty}[V^*(x) - \hat{Q}_{n_k}(x,a)]^2\right] \\ &\geq \mathbb{E}\left[\liminf_{k\to\infty}[V^*(x) - \hat{Q}_k(x,a)]^2\right] \geq \mathbb{E}[\hat{A}(x,a)^2],\end{aligned}$$

where the interchange of limit and expectation in the first equality is due the uniform boundedness of $\tilde{Q}_k(x,a)$, the second equality is due to the fact that $V_{n_k}(x)$ converges to $V^*(x)$ a.s., the first inequality follows from $\mathbb{V}\text{ar}[\hat{Q}_k] \leq \mathbb{V}\text{ar}[\tilde{Q}_k]$, the second inequality is due to Fatou's Lemma, and the last inequality is a basic property of limit inferior. Since this result holds true for any converging subsequence of $\tilde{Q}_{n_k}(x,a)$, with the one that achieves $\tilde{A}(x,a)$ being one of them, we can conclude that $\mathbb{E}[\tilde{A}(x,a)^2] \geq \mathbb{E}[\hat{A}(x,a)^2]$.

B Additional Experimental Results

In this section we provide additional experimental results that expand upon those provided in the main paper.

The full set of experimental results during the training phase for Acrobot, Mountain Car, and Lunar Lander are presented in Figures 3, 4, and 5, respectively.

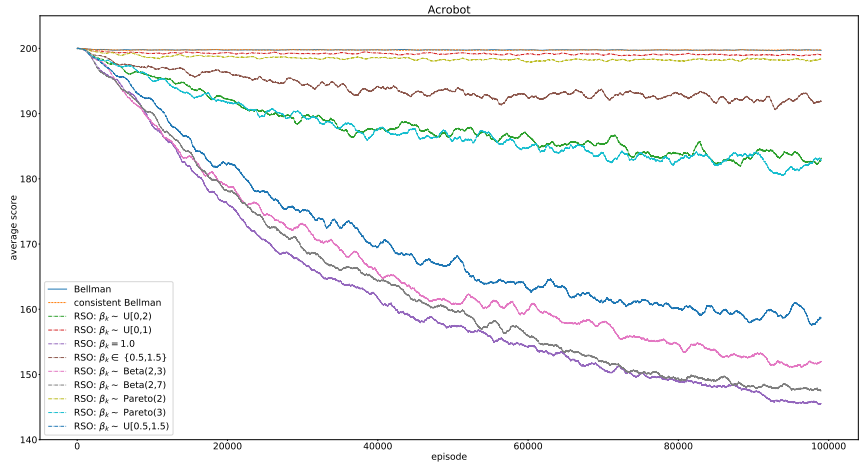


Figure 3: Average number of steps needed to solve Acrobot minimization problem during training phase. Full set of experiments under all RSO instances.

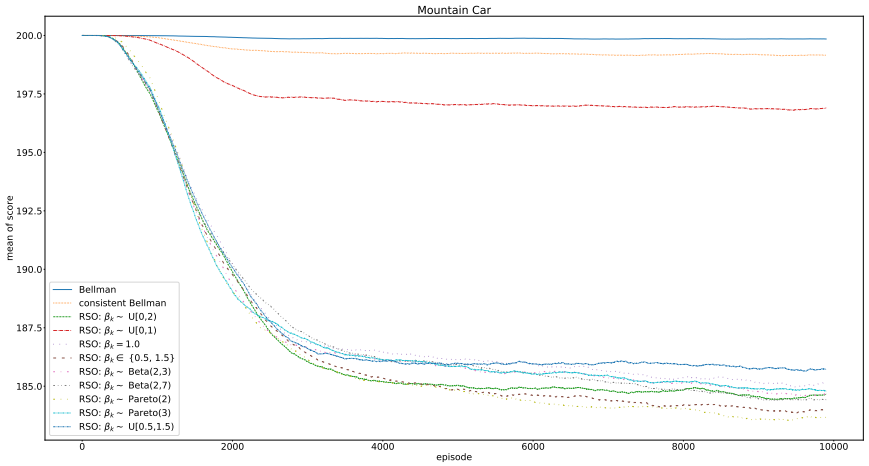


Figure 4: Average number of steps needed to solve Mountain Car minimization problem during training phase. Full set of experiments under all RSO instances.

The statistics for each problem during the training phase are presented in Table 1, while the statistics for each problem during the testing phase are presented in Table 2.

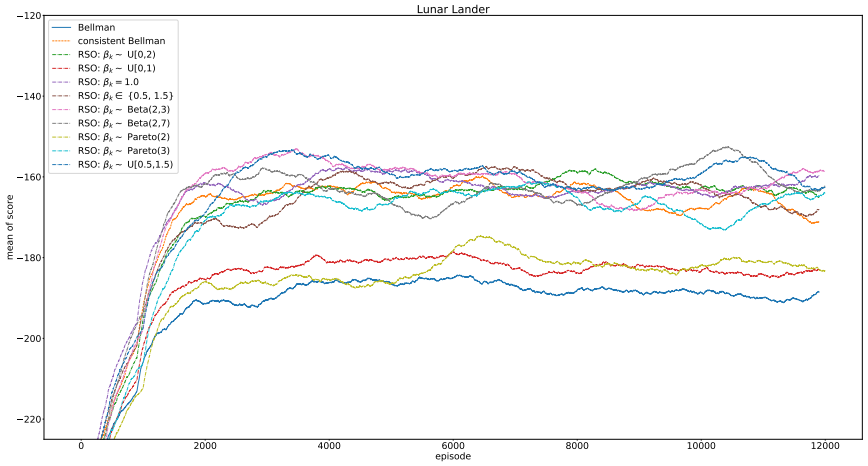


Figure 5: Average score in solving Lunar Lander maximization problem during training phase. Full set of experiments under all RSO instances.

Training Score	Acrobot	Mountain Car	Cartpole	Lunar Lander
Bellman	$199.7 \pm 0.02\%$	$199.8 \pm 0.01\%$	$184.2 \pm 0.32\%$	$-188.5 \pm 0.84\%$
Consistent Bellman	$199.8 \pm 0.01\%$	$199.2 \pm 0.02\%$	$189.2 \pm 0.22\%$	$-171.1 \pm 0.95\%$
$\beta_k \sim U[0, 2)$	$184.1 \pm 0.14\%$	$184.5 \pm 0.09\%$	$189.3 \pm 0.21\%$	$-165.0 \pm 1.00\%$
$\beta_k \sim U[0, 1)$	$198.9 \pm 0.03\%$	$196.8 \pm 0.04\%$	$186.1 \pm 0.25\%$	$-183.0 \pm 0.87\%$
$\beta_k = 1.0$	$\mathbf{145.7} \pm 0.26\%$	$184.8 \pm 0.09\%$	$\mathbf{190.9} \pm 0.19\%$	*-159.7* $\pm 1.05\%$
$\beta_k \in \{0.5, 1.5\}$	$192.1 \pm 0.10\%$	*183.9* $\pm 0.09\%$	$185.8 \pm 0.26\%$	$-162.7 \pm 1.05\%$
$\beta_k \sim U[0.5, 1.5)$	$161.0 \pm 0.22\%$	$185.7 \pm 0.09\%$	$189.6 \pm 0.21\%$	$-168.2 \pm 0.98\%$
$\beta_k \sim$ Beta(2,3)	$151.3 \pm 0.24\%$	$184.6 \pm 0.09\%$	$187.1 \pm 0.26\%$	$\mathbf{-158.5} \pm 1.05\%$
$\beta_k \sim$ Beta(2,7)	*147.7* $\pm 0.25\%$	$184.4 \pm 0.09\%$	*190.4* $\pm 0.20\%$	$-162.3 \pm 1.05\%$
$\beta_k \sim$ Pareto(2)	$198.4 \pm 0.04\%$	$\mathbf{183.7} \pm 0.09\%$	$184.3 \pm 0.32\%$	$-183.2 \pm 0.90\%$
$\beta_k \sim$ Pareto(3)	$181.4 \pm 0.16\%$	$184.9 \pm 0.09\%$	$185.6 \pm 0.28\%$	$-163.9 \pm 1.01\%$

Table 1: Mean scores for solving each problem during training phase. Full set of experiments under all RSO instances. The best scores are highlighted in bold and the second best scores are highlighted in italics.

Testing Score	Acrobot	Mountain Car	Cartpole	Lunar Lander
Bellman	$189.1 \pm 0.17\%$	$131.2 \pm 0.23\%$	$189.2 \pm 0.24\%$	$-231.0 \pm 0.92\%$
Consistent Bellman	$185.3 \pm 0.20\%$	$127.2 \pm 0.22\%$	$185.5 \pm 0.28\%$	$-185.1 \pm 0.98\%$
$\beta_k \sim U[0, 2)$	$189.5 \pm 0.16\%$	*121.2* $\pm 0.21\%$	*189.6* $\pm 0.23\%$	$-164.4 \pm 1.05\%$
$\beta_k \sim U[0, 1)$	$184.9 \pm 0.18\%$	$126.9 \pm 0.23\%$	$184.9 \pm 0.26\%$	$-207.0 \pm 0.94\%$
$\beta_k = 1.0$	$189.2 \pm 0.18\%$	$121.9 \pm 0.21\%$	$189.6 \pm 0.25\%$	$\mathbf{-157.8} \pm 1.10\%$
$\beta_k \in \{0.5, 1.5\}$	*181.3* $\pm 0.23\%$	$122.3 \pm 0.20\%$	$181.6 \pm 0.33\%$	$-174.0 \pm 1.01\%$
$\beta_k \sim U[0.5, 1.5)$	$192.4 \pm 0.13\%$	$122.8 \pm 0.21\%$	$\mathbf{192.8} \pm 0.18\%$	$-168.1 \pm 1.08\%$
$\beta_k \sim$ Beta(2,3)	$185.0 \pm 0.20\%$	$122.6 \pm 0.21\%$	$185.0 \pm 0.29\%$	*-163.5* $\pm 1.13\%$
$\beta_k \sim$ Beta(2,7)	$186.2 \pm 0.19\%$	$122.3 \pm 0.21\%$	$186.4 \pm 0.27\%$	$-164.8 \pm 1.06\%$
$\beta_k \sim$ Pareto(2)	$\mathbf{180.7} \pm 0.37\%$	$125.0 \pm 0.20\%$	$180.1 \pm 0.52\%$	$-216.9 \pm 0.94\%$
$\beta_k \sim$ Pareto(3)	$186.6 \pm 0.21\%$	$\mathbf{121.1} \pm 0.21\%$	$186.5 \pm 0.29\%$	$-166.2 \pm 1.04\%$

Table 2: Mean scores for solving each problem during testing phase. Full set of experiments under all RSO instances. The best scores are highlighted in bold and the second best scores are highlighted in italics.

C Python Code

We tested the various operators of interest on several RL problems and algorithms. For our empirical comparisons, the existing code that updates the Q-learning value based on the Bellman operator $\mathcal{T}_B$ is replaced with the corresponding code for the $\mathcal{T}_C$ and $\mathcal{T}_{\beta_k}$ operators. In particular, the snippets of code in Figure 6 describe how this is generically implemented for the original $\mathcal{T}_B$ operator together with the added $\mathcal{T}_C$ and $\mathcal{T}_{\beta_k}$ operators, respectively.

```
def UpdateQBellman(self,currentState,action,nextState,reward,alpha,gamma):
  Qvalue=self.Q[currentState,action]
  rvalue=reward+gamma*max([self.Q[nextState,a] for a in self.actionsSet])
  self.Q[currentState,action] += alpha*(rvalue - Qvalue)

def UpdateQConsistent(self,currentState,action,nextState,reward,alpha,gamma):
  Qvalue=self.Q[currentState,action]
  rvalue=reward+gamma*(max([self.Q[nextState,a] for a in self.actionsSet])
      if currentState != nextState else Qvalue)
  self.Q[currentState,action] += alpha*(rvalue - Qvalue)

def UpdateQRSO(self,currentState,action,nextState,reward,alpha,gamma,beta):
  Qvalue=self.Q[currentState,action]
  rvalue=reward+(gamma*(max([self.Q[nextState,a] for a in self.actionsSet]))
      -beta*(max([self.Q[currentState,a] for a in self.actionsSet])-Qvalue))
  self.Q[currentState,action] += alpha*(rvalue - Qvalue)
```

Figure 6: Generic python code for all three operators