Actor-Critic Algorithms for Risk-Sensitive MDPs

Prashanth L.A., Mohammad Ghavamzadeh

Actor-Critic Algorithms for Risk-Sensitive MDPs

Prashanth L.A.
INRIA Lille - Team SequeL

Mohammad Ghavamzadeh*
INRIA Lille - Team SequeL & Adobe Research

Abstract

In many sequential decision-making problems we may want to manage risk by minimizing some measure of variability in rewards in addition to maximizing a standard criterion. Variance-related risk measures are among the most common risk-sensitive criteria in finance and operations research. However, optimizing many such criteria is known to be a hard problem. In this paper, we consider both discounted and average reward Markov decision processes. For each formulation, we first define a measure of variability for a policy, which in turn gives us a set of risk-sensitive criteria to optimize. For each of these criteria, we derive a formula for computing its gradient. We then devise actor-critic algorithms for estimating the gradient and updating the policy parameters in the ascent direction. We establish the convergence of our algorithms to locally risk-sensitive optimal policies. Finally, we demonstrate the usefulness of our algorithms in a traffic signal control application.

1 Introduction

The usual optimization criteria for an infinite horizon Markov decision process (MDP) are the *expected sum of discounted rewards* and the *average reward*. Many algorithms have been developed to maximize these criteria both when the model of the system is known (planning) and unknown (learning). These algorithms can be categorized to value function based methods that are mainly based on the two celebrated dynamic programming algorithms *value iteration* and *policy iteration*; and policy gradient methods that are based on updating the policy parameters in the direction of the gradient of a performance measure (the value function of the initial state or the average reward). However in many applications, we may prefer to minimize some measure of *risk* as well as maximizing a usual optimization criterion. In such cases, we would like to use a criterion that incorporates a penalty for the *variability* induced by a given policy. This variability can be due to two types of uncertainties: **1)** uncertainties in the model parameters, which is the topic of *robust* MDPs (e.g., [12, 7, 23]), and **2)** the inherent uncertainty related to the stochastic nature of the system, which is the topic of *risk-sensitive* MDPs (e.g., [10]).

In risk-sensitive sequential decision-making, the objective is to maximize a risk-sensitive criterion such as the expected exponential utility [10], a variance-related measure [18, 8], or the percentile performance [9]. The issue of how to construct such criteria in a manner that will be both conceptually meaningful and mathematically tractable is still an open question. Although risk-sensitive sequential decision-making has a long history in operations research and finance, it has only recently grabbed attention in the machine learning community. This is why most of the work on this topic (including those mentioned above) has been in the context of MDPs (when the model is known) and much less work has been done within the reinforcement learning (RL) framework. In risk-sensitive RL, we can mention the work by Borkar [4, 5] who considered the expected exponential utility and the one by Tamar et al. [21] on several variance-related measures. Tamar et al. [21] study stochastic shortest path problems, and in this context, propose a policy gradient algorithm for maximizing several risk-sensitive criteria that involve both the expectation and variance of the *return* random variable (defined as the sum of rewards received in an episode).

*Mohammad Ghavamzadeh is at Adobe Research, on leave from INRIA Lille - Team SequeL.

In this paper, we develop actor-critic algorithms for optimizing variance-related risk measures in both discounted and average reward MDPs. Our contributions can be summarized as follows:
• In the discounted reward setting we define the measure of variability as the *variance of the return* (similar to [21]). We formulate a constrained optimization problem with the aim of maximizing the mean of the return subject to its variance being bounded from above. We employ the Lagrangian relaxation procedure [1] and derive a formula for the gradient of the Lagrangian. Since this requires the gradient of the value function at every state of the MDP (see the discussion in Sections 3 and 4), we estimate the gradient of the Lagrangian using two simultaneous perturbation methods: *simultaneous perturbation stochastic approximation* (SPSA) [19] and *smoothed functional* (SF) [11], resulting in two separate discounted reward actor-critic algorithms.[1]
• In the average reward formulation, we first define the measure of variability as the *long-run variance* of a policy, and using a constrained optimization problem similar to the discounted case, derive an expression for the gradient of the Lagrangian. We then develop an actor-critic algorithm with compatible features [20, 13] to estimate the gradient and to optimize the policy parameters.
• Using the ordinary differential equations (ODE) approach, we establish the asymptotic convergence of our algorithms to locally risk-sensitive optimal policies. Further, we demonstrate the usefulness of our algorithms in a traffic signal control problem.
In comparison to [21], which is the closest related work, we would like to remark that while the authors there develop policy gradient methods for stochastic shortest path problems, we devise actor-critic algorithms for both discounted and average reward settings. Moreover, we note the difficulty in the discounted formulation that requires to estimate the gradient of the value function at every state of the MDP, and thus, motivated us to employ simultaneous perturbation techniques.

2 Preliminaries

We consider problems in which the agent's interaction with the environment is modeled as a MDP. A MDP is a tuple $(\mathcal{X}, \mathcal{A}, R, P, P_0)$ where $\mathcal{X} = \{1, \ldots, n\}$ and $\mathcal{A} = \{1, \ldots, m\}$ are the state and action spaces; $R(x, a)$ is the reward random variable whose expectation is denoted by $r(x, a) = \mathbb{E}\big[R(x, a)\big]$; $P(\cdot|x, a)$ is the transition probability distribution; and $P_0(\cdot)$ is the initial state distribution. We also need to specify the rule according to which the agent selects actions at each state. A *stationary policy* $\mu(\cdot|x)$ is a probability distribution over actions, conditioned on the current state. In policy gradient and actor-critic methods, we define a class of parameterized stochastic policies $\big\{\mu(\cdot|x;\theta), x \in \mathcal{X}, \theta \in \Theta \subseteq \mathbb{R}^{\kappa_1}\big\}$, estimate the gradient of a performance measure w.r.t. the policy parameters θ from the observed system trajectories, and then improve the policy by adjusting its parameters in the direction of the gradient. Since in this setting a policy μ is represented by its κ_1-dimensional parameter vector θ, policy dependent functions can be written as a function of θ in place of μ. So, we use μ and θ interchangeably in the paper.

We denote by $d^\mu(x)$ and $\pi^\mu(x, a) = d^\mu(x)\mu(a|x)$ the stationary distribution of state x and state-action pair (x, a) under policy μ, respectively. In the discounted formulation, we also define the discounted visiting distribution of state x and state-action pair (x, a) under policy μ as $d^\mu_\gamma(x|x^0) = (1-\gamma)\sum_{t=0}^\infty \gamma^t \Pr(x_t = x|x_0 = x^0; \mu)$ and $\pi^\mu_\gamma(x, a|x^0) = d^\mu_\gamma(x|x^0)\mu(a|x)$.

3 Discounted Reward Setting

For a given policy μ, we define the return of a state x (state-action pair (x, a)) as the sum of discounted rewards encountered by the agent when it starts at state x (state-action pair (x, a)) and then follows policy μ, i.e.,

$$D^\mu(x) = \sum_{t=0}^\infty \gamma^t R(x_t, a_t) \mid x_0 = x,\ \mu, \qquad D^\mu(x, a) = \sum_{t=0}^\infty \gamma^t R(x_t, a_t) \mid x_0 = x,\ a_0 = a,\ \mu.$$

The expected value of these two random variables are the value and action-value functions of policy μ, i.e., $V^\mu(x) = \mathbb{E}\big[D^\mu(x)\big]$ and $Q^\mu(x, a) = \mathbb{E}\big[D^\mu(x, a)\big]$. The goal in the standard discounted reward formulation is to find an optimal policy $\mu^* = \arg\max_\mu V^\mu(x^0)$, where x^0 is the initial state of the system. This can be easily extended to the case that the system has more than one initial state $\mu^* = \arg\max_\mu \sum_{x \in \mathcal{X}} P_0(x) V^\mu(x)$.

[1] We note here that our algorithms can be easily extended to other variance-related risk criteria such as the Sharpe ratio, which is popular in financial decision-making [17] (see Appendix D in the supporting material).

The most common measure of the *variability* in the stream of rewards is the *variance of the return*

$$\Lambda^\mu(x) = \mathbb{E}\big[D^\mu(x)^2\big] - V^\mu(x)^2 = U^\mu(x) - V^\mu(x)^2, \tag{1}$$

first introduced by Sobel [18]. Note that $U^\mu(x) \stackrel{\triangle}{=} \mathbb{E}\left[D^\mu(x)^2\right]$ is the *square reward value function* of state x under policy μ. Although Λ^μ of (1) satisfies a Bellman equation, unfortunately, it lacks the monotonicity property of dynamic programming (DP), and thus, it is not clear how the related risk measures can be optimized by standard DP algorithms [18]. This is why policy gradient and actor-critic algorithms are good candidates to deal with this risk measure. We consider the following risk-sensitive measure for discounted MDPs: for a given $\alpha > 0$,

$$\max_\theta V^\theta(x^0) \qquad \text{subject to} \qquad \Lambda^\theta(x^0) \leq \alpha. \tag{2}$$

To solve (2), we employ the Lagrangian relaxation procedure [1] to convert it to the following unconstrained problem:

$$\max_\lambda \min_\theta \Big(L(\theta,\lambda) \stackrel{\triangle}{=} -V^\theta(x^0) + \lambda\big(\Lambda^\theta(x^0) - \alpha\big)\Big), \tag{3}$$

where λ is the Lagrange multiplier. The goal here is to find the saddle point of $L(\theta,\lambda)$, i.e., a point (θ^*,λ^*) that satisfies $L(\theta,\lambda^*) \geq L(\theta^*,\lambda^*) \geq L(\theta^*,\lambda), \forall\theta, \forall\lambda > 0$. This is achieved by descending in θ and ascending in λ using the gradients $\nabla_\theta L(\theta,\lambda) = -\nabla_\theta V^\theta(x^0) + \lambda\nabla_\theta\Lambda^\theta(x^0)$ and $\nabla_\lambda L(\theta,\lambda) = \Lambda^\theta(x^0) - \alpha$, respectively. Since $\nabla\Lambda^\theta(x^0) = \nabla U^\theta(x^0) - 2V^\theta(x^0)\nabla V^\theta(x^0)$, in order to compute $\nabla\Lambda^\theta(x^0)$, we need to calculate $\nabla U^\theta(x^0)$ and $\nabla V^\theta(x^0)$. From the Bellman equation of $\Lambda^\mu(x)$, proposed by Sobel [18], it is straightforward to derive Bellman equations for $U^\mu(x)$ and the *square reward action-value function* $W^\mu(x,a) \stackrel{\triangle}{=} \mathbb{E}\big[D^\mu(x,a)^2\big]$ (see Appendix B.1). Using these definitions and notations we are now ready to derive expressions for the gradient of $V^\theta(x^0)$ and $U^\theta(x^0)$ that are the main ingredients in calculating $\nabla_\theta L(\theta,\lambda)$.

Lemma 1 *Assuming for all (x,a), $\mu(a|x;\theta)$ is continuously differentiable in θ, we have*

$$\begin{aligned}
(1-\gamma)\nabla V^\theta(x^0) &= \sum_{x,a} \pi_\gamma^\theta(x,a|x^0)\nabla\log\mu(a|x;\theta)Q^\theta(x,a),\\
(1-\gamma^2)\nabla U^\theta(x^0) &= \sum_{x,a} \widetilde{\pi}_\gamma^\theta(x,a|x^0)\nabla\log\mu(a|x;\theta)W^\theta(x,a)\\
&\quad + 2\gamma\sum_{x,a,x'} \widetilde{\pi}_\gamma^\theta(x,a|x^0)P(x'|x,a)r(x,a)\nabla V^\theta(x'),
\end{aligned}$$

where $\widetilde{\pi}_\gamma^\theta(x,a|x^0) = \widetilde{d}_\gamma^\theta(x|x^0)\mu(a|x)$ and $\widetilde{d}_\gamma^\theta(x|x^0) = (1-\gamma^2)\sum_{t=0}^\infty \gamma^{2t}\Pr(x_t = x|x_0 = x^0;\theta)$.

The proof of the above lemma is available in Appendix B.2. It is challenging to devise an efficient method to estimate $\nabla_\theta L(\theta,\lambda)$ using the gradient formulas of Lemma 1. This is mainly because **1)** two different sampling distributions (π_γ^θ and $\widetilde{\pi}_\gamma^\theta$) are used for $\nabla V^\theta(x^0)$ and $\nabla U^\theta(x^0)$, and **2)** $\nabla V^\theta(x')$ appears in the second sum of $\nabla U^\theta(x^0)$ equation, which implies that we need to estimate the gradient of the value function V^θ at every state of the MDP. These are the main motivations behind using simultaneous perturbation methods for estimating $\nabla_\theta L(\theta,\lambda)$ in Section 4.

4 Discounted Reward Algorithms

In this section, we present actor-critic algorithms for optimizing the risk-sensitive measure (2) that are based on two simultaneous perturbation methods: *simultaneous perturbation stochastic approximation* (SPSA) and *smoothed functional* (SF) [3]. The idea in these methods is to estimate the gradients $\nabla V^\theta(x^0)$ and $\nabla U^\theta(x^0)$ using two simulated trajectories of the system corresponding to policies with parameters θ and $\theta^+ = \theta + \beta\Delta$. Here $\beta > 0$ is a positive constant and Δ is a perturbation random variable, i.e., a κ_1-vector of independent Rademacher (for SPSA) and Gaussian $\mathcal{N}(0,1)$ (for SF) random variables. In our actor-critic algorithms, the critic uses linear approximation for the value and square value functions, i.e., $\widehat{V}(x) \approx v^\top\phi_v(x)$ and $\widehat{U}(x) \approx u^\top\phi_u(x)$, where the features $\phi_v(\cdot)$ and $\phi_u(\cdot)$ are from low-dimensional spaces $\mathbb{R}^{\kappa_2}$ and $\mathbb{R}^{\kappa_3}$, respectively.

SPSA-based gradient estimates were first proposed in [19] and have been widely studied and found to be highly efficient in various settings, especially those involving high-dimensional parameters. The SPSA-based estimate for $\nabla V^\theta(x^0)$, and similarly for $\nabla U^\theta(x^0)$, is given by:

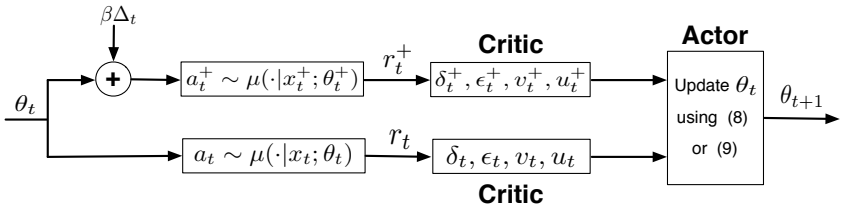


Figure 1: The overall flow of our simultaneous perturbation based actor-critic algorithms.

$$\partial_{\theta^{(i)}} \widehat{V}^{\theta}(x^0) \quad \approx \quad \frac{\widehat{V}^{\theta+\beta\Delta}(x^0) - \widehat{V}^{\theta}(x^0)}{\beta\Delta^{(i)}}, \qquad i = 1, \ldots, \kappa_1, \tag{4}$$

where Δ is a vector of independent Rademacher random variables. The advantage of this estimator is that it perturbs all directions at the same time (the numerator is identical in all κ_1 components). So, the number of function measurements needed for this estimator is always two, independent of the dimension κ_1. However, unlike the SPSA estimates in [19] that use two-sided balanced estimates (simulations with parameters $\theta - \beta\Delta$ and $\theta + \beta\Delta$), our gradient estimates are one-sided (simulations with parameters θ and $\theta + \beta\Delta$) and resemble those in [6]. The use of one-sided estimates is primarily because the updates of the Lagrangian parameter λ require a simulation with the running parameter θ. Using a balanced gradient estimate would therefore come at the cost of an additional simulation (the resulting procedure would then require three simulations), which we avoid by using one-sided gradient estimates.

SF-based method estimates not the gradient of a function $H(\theta)$ itself, but rather the convolution of $\nabla H(\theta)$ with the Gaussian density function $\mathcal{N}(\mathbf{0}, \beta^2 \boldsymbol{I})$, i.e.,

$$C_\beta H(\theta) = \int \mathcal{G}_\beta(\theta - z) \nabla_z H(z) dz = \int \nabla_z \mathcal{G}_\beta(z) H(\theta - z) dz = \frac{1}{\beta} \int -z' \mathcal{G}_1(z') H(\theta - \beta z') dz',$$

where $\mathcal{G}_\beta$ is a κ_1-dimensional p.d.f. The first equality above follows by using integration by parts and the second one by using the fact that $\nabla_z \mathcal{G}_\beta(z) = \frac{-z}{\beta^2} \mathcal{G}_\beta(z)$ and by substituting $z' = z/\beta$. As $\beta \to 0$, it can be seen that $C_\beta H(\theta)$ converges to $\nabla_\theta H(\theta)$ (see Chapter 6 of [3]). Thus, a one-sided SF estimate of $\nabla V^\theta(x^0)$ is given by

$$\partial_{\theta^{(i)}} \widehat{V}^{\theta}(x^0) \quad \approx \quad \frac{\Delta^{(i)}}{\beta} \left(\widehat{V}^{\theta+\beta\Delta}(x^0) - \widehat{V}^{\theta}(x^0) \right), \qquad i = 1, \ldots, \kappa_1, \tag{5}$$

where Δ is a vector of independent Gaussian $\mathcal{N}(0, 1)$ random variables.

The overall flow of our proposed actor-critic algorithms is illustrated in Figure 1 and involves the following main steps at each time step t:
(1) Take action $a_t \sim \mu(\cdot|x_t;\theta_t)$, observe the reward $r(x_t, a_t)$ and next state x_{t+1} in the first trajectory.
(2) Take action $a_t^+ \sim \mu(\cdot|x_t^+;\theta_t^+)$, observe the reward $r(x_t^+, a_t^+)$ and next state x_{t+1}^+ in the second trajectory.
(3) Critic Update: Calculate the temporal difference (TD)-errors δ_t, δ_t^+ for the value and ϵ_t, ϵ_t^+ for the square value functions using (7), and update the critic parameters v_t, v_t^+ for the value and u_t, u_t^+ for the square value functions as follows:

$$\begin{aligned} v_{t+1} &= v_t + \zeta_3(t)\delta_t \phi_v(x_t), & v_{t+1}^+ &= v_t^+ + \zeta_3(t)\delta_t^+ \phi_v(x_t^+), \\ u_{t+1} &= u_t + \zeta_3(t)\epsilon_t \phi_u(x_t), & u_{t+1}^+ &= u_t^+ + \zeta_3(t)\epsilon_t^+ \phi_u(x_t^+), \end{aligned} \tag{6}$$

where the TD-errors $\delta_t, \delta_t^+, \epsilon_t, \epsilon_t^+$ in (6) are computed as

$$\begin{aligned} \delta_t &= r(x_t, a_t) + \gamma v_t^\top \phi_v(x_{t+1}) - v_t^\top \phi_v(x_t), & \delta_t^+ &= r(x_t^+, a_t^+) + \gamma v_t^{+\top} \phi_v(x_{t+1}^+) - v_t^{+\top} \phi_v(x_t^+), \\ \epsilon_t &= r(x_t, a_t)^2 + 2\gamma r(x_t, a_t) v_t^\top \phi_v(x_{t+1}) + \gamma^2 u_t^\top \phi_u(x_{t+1}) - u_t^\top \phi_u(x_t), \\ \epsilon_t^+ &= r(x_t^+, a_t^+)^2 + 2\gamma r(x_t^+, a_t^+) v_t^{+\top} \phi_v(x_{t+1}^+) + \gamma^2 u_t^{+\top} \phi_u(x_{t+1}^+) - u_t^{+\top} \phi_u(x_t^+). \end{aligned} \tag{7}$$

This TD algorithm to learn the value and square value functions is a straightforward extension of the algorithm proposed by Tamar et al. [22] to the discounted setting. Note that the TD-error ϵ for the square value function U comes directly from the Bellman equation for U (see Appendix B.1).
(4) Actor Update: Estimate the gradients $\nabla V^\theta(x^0)$ and $\nabla U^\theta(x^0)$ using SPSA (4) or SF (5) and update the policy parameter θ and the Lagrange multiplier λ as follows: For $i = 1, \ldots, \kappa_1$,

$$\theta_{t+1}^{(i)} = \Gamma_i\left[\theta_t^{(i)} + \frac{\zeta_2(t)}{\beta\Delta_t^{(i)}}\Big(\big(1+2\lambda_t v_t^\top\phi_v(x^0)\big)(v_t^+ - v_t)^\top\phi_v(x^0) - \lambda_t(u_t^+ - u_t)^\top\phi_u(x^0)\Big)\right], \textbf{SPSA} \quad (8)$$

$$\theta_{t+1}^{(i)} = \Gamma_i\left[\theta_t^{(i)} + \frac{\zeta_2(t)\Delta_t^{(i)}}{\beta}\Big(\big(1+2\lambda_t v_t^\top\phi_v(x^0)\big)(v_t^+ - v_t)^\top\phi_v(x^0) - \lambda_t(u_t^+ - u_t)^\top\phi_u(x^0)\Big)\right], \textbf{SF} \quad (9)$$

$$\lambda_{t+1} = \Gamma_\lambda\left[\lambda_t + \zeta_1(t)\Big(u_t^\top\phi_u(x^0) - \big(v_t^\top\phi_v(x^0)\big)^2 - \alpha\Big)\right]. \quad (10)$$

Note that **1)** the λ-update is the same for both SPSA and SF methods, **2)** $\Delta_t^{(i)}$'s are independent Rademacher and Gaussian $\mathcal{N}(0,1)$ random variables in SPSA and SF updates, respectively, **3)** Γ is an operator that projects a vector $\theta \in \mathbb{R}^{\kappa_1}$ to the closest point in a compact and convex set $C \subset \mathbb{R}^{\kappa_1}$, and Γ_λ is a projection operator to $[0, \lambda_{\max}]$. These projection operators are necessary to ensure convergence of the algorithms, and **4)** the step-size schedules $\{\zeta_3(t)\}$, $\{\zeta_2(t)\}$, and $\{\zeta_1(t)\}$ are chosen such that the critic updates are on the fastest time-scale, the policy parameter update is on the intermediate time-scale, and the Lagrange multiplier update is on the slowest time-scale (see Appendix A in the supplementary material for the conditions on the step-size schedules). A proof of convergence of the SPSA and SF algorithms to a (local) saddle point of the risk-sensitive objective function $\widehat{L}(\theta,\lambda) \stackrel{\triangle}{=} -\widehat{V}^\theta(x^0) + \lambda(\widehat{\Lambda}^\theta(x^0) - \alpha)$ is given in Appendix B.3.

5 Average Reward Setting

The average reward per step under policy μ is defined as (see Sec. 2 for the definitions of d^μ and π^μ)

$$\rho(\mu) \;=\; \lim_{T\to\infty}\frac{1}{T}\mathbb{E}\left[\sum_{t=0}^{T-1} R_t \mid \mu\right] \;=\; \sum_{x,a} d^\mu(x)\mu(a|x)r(x,a).$$

The goal in the standard (risk-neutral) average reward formulation is to find an *average optimal* policy, i.e., $\mu^* = \arg\max_\mu \rho(\mu)$. Here a policy μ is assessed according to the expected differential reward associated with states or state-action pairs. For all states $x \in \mathcal{X}$ and actions $a \in \mathcal{A}$, the *differential* action-value and value functions of policy μ are defined as

$$Q^\mu(x,a) = \sum_{t=0}^{\infty}\mathbb{E}\big[R_t - \rho(\mu) \mid x_0 = x, a_0 = a, \mu\big], \qquad V^\mu(x) = \sum_a \mu(a|x)Q^\mu(x,a).$$

In the context of risk-sensitive MDPs, different criteria have been proposed to define a measure of *variability*, among which we consider the *long-run variance* of μ [8] defined as

$$\Lambda(\mu) \;=\; \sum_{x,a}\pi^\mu(x,a)\big[r(x,a) - \rho(\mu)\big]^2 \;=\; \lim_{T\to\infty}\frac{1}{T}\mathbb{E}\left[\sum_{t=0}^{T-1}\big(R_t - \rho(\mu)\big)^2 \mid \mu\right]. \quad (11)$$

This notion of variability is based on the observation that it is the frequency of occurrence of state-action pairs that determine the variability in the average reward. It is easy to show that

$$\Lambda(\mu) = \eta(\mu) - \rho(\mu)^2, \qquad \text{where} \quad \eta(\mu) = \sum_{x,a}\pi^\mu(x,a)r(x,a)^2.$$

We consider the following risk-sensitive measure for average reward MDPs in this paper:

$$\max_\theta \rho(\theta) \qquad \text{subject to} \qquad \Lambda(\theta) \le \alpha, \quad (12)$$

for a given $\alpha > 0$. As in the discounted setting, we employ the Lagrangian relaxation procedure to convert (12) to the unconstrained problem

$$\max_\lambda \min_\theta \Big(L(\theta,\lambda) \stackrel{\triangle}{=} -\rho(\theta) + \lambda\big(\Lambda(\theta) - \alpha\big)\Big).$$

Similar to the discounted case, we descend in θ using $\nabla_\theta L(\theta,\lambda) = -\nabla_\theta\rho(\theta) + \lambda\nabla_\theta\Lambda(\theta)$ and ascend in λ using $\nabla_\lambda L(\theta,\lambda) = \Lambda(\theta) - \alpha$, to find the saddle point of $L(\theta,\lambda)$. Since $\nabla\Lambda(\theta) = \nabla\eta(\theta) -$

$2\rho(\theta)\nabla\rho(\theta)$, in order to compute $\nabla\Lambda(\theta)$ it would be enough to calculate $\nabla\eta(\theta)$. Let U^μ and W^μ denote the differential value and action-value functions associated with the square reward under policy μ, respectively. These two quantities satisfy the following Poisson equations:

$$\eta(\mu) + U^\mu(x) = \sum_a \mu(a|x)\big[r(x,a)^2 + \sum_{x'} P(x'|x,a)U^\mu(x')\big],$$
$$\eta(\mu) + W^\mu(x,a) = r(x,a)^2 + \sum_{x'} P(x'|x,a)U^\mu(x'). \tag{13}$$

We calculate the gradients of $\rho(\theta)$ and $\eta(\theta)$ as (see Lemma 5 in Appendix C.1 in the supplementary material):

$$\nabla\rho(\theta) = \sum_{x,a} \pi(x,a;\theta)\nabla\log\mu(a|x;\theta)Q(x,a;\theta), \tag{14}$$

$$\nabla\eta(\theta) = \sum_{x,a} \pi(x,a;\theta)\nabla\log\mu(a|x;\theta)W(x,a;\theta). \tag{15}$$

Note that (15) for calculating $\nabla\eta(\theta)$ has close resemblance to (14) for $\nabla\rho(\theta)$, and thus, similar to what we have for (14), any function $b : \mathcal{X} \to \mathbb{R}$ can be added or subtracted to $W(x,a;\theta)$ on the RHS of (15) without changing the result of the integral (see e.g., [2]). So, we can replace $W(x,a;\theta)$ with the square reward advantage function $B(x,a;\theta) = W(x,a;\theta) - U(x;\theta)$ on the RHS of (15) in the same manner as we can replace $Q(x,a;\theta)$ with the advantage function $A(x,a;\theta) = Q(x,a;\theta) - V(x;\theta)$ on the RHS of (14) without changing the result of the integral. We define the temporal difference (TD) errors δ_t and ϵ_t for the differential value and square value functions as

$$\delta_t = R(x_t,a_t) - \widehat{\rho}_{t+1} + \widehat{V}(x_{t+1}) - \widehat{V}(x_t), \qquad \epsilon_t = R(x_t,a_t)^2 - \widehat{\eta}_{t+1} + \widehat{U}(x_{t+1}) - \widehat{U}(x_t).$$

If $\widehat{V}$, $\widehat{U}$, $\widehat{\rho}$, and $\widehat{\eta}$ are unbiased estimators of V^μ, U^μ, $\rho(\mu)$, and $\eta(\mu)$, respectively, then we can show that δ_t and ϵ_t are unbiased estimates of the advantage functions A^μ and B^μ, i.e., $\mathbb{E}[\,\delta_t|\,x_t,a_t,\mu] = A^\mu(x_t,a_t)$, and $\mathbb{E}[\,\epsilon_t|\,x_t,a_t,\mu] = B^\mu(x_t,a_t)$ (see Lemma 6 in Appendix C.2). From this, we notice that $\delta_t\psi_t$ and $\epsilon_t\psi_t$ are unbiased estimates of $\nabla\rho(\mu)$ and $\nabla\eta(\mu)$, respectively, where $\psi_t = \psi(x_t,a_t) = \nabla\log\mu(a_t|x_t)$ is the *compatible* feature (see e.g., [20, 13]).

6 Average Reward Algorithm

We now present our risk-sensitive actor-critic algorithm for average reward MDPs. Algorithm 1 presents the complete structure of the algorithm along with update rules for the average rewards $\widehat{\rho}_t, \widehat{\eta}_t$; TD errors δ_t, ϵ_t; critic v_t, u_t; and actor θ_t, λ_t parameters. The projection operators Γ and Γ_λ are as defined in Section 4, and similar to the discounted setting, are necessary for the convergence proof of the algorithm. The step-size schedules satisfy the standard conditions for stochastic approximation algorithms, and ensure that the average and critic updates are on the (same) fastest time-scale $\{\zeta_4(t)\}$ and $\{\zeta_3(t)\}$, the policy parameter update is on the intermediate time-scale $\{\zeta_2(t)\}$, and the Lagrange multiplier is on the slowest time-scale $\{\zeta_1(t)\}$ (see Appendix A). This results in a three time-scale stochastic approximation algorithm. As in the discounted setting, the critic uses linear approximation for the differential value and square value functions, i.e., $\widehat{V}(x) = v^\top\phi_v(x)$ and $\widehat{U}(x) = u^\top\phi_u(x)$, where $\phi_v(\cdot)$ and $\phi_u(\cdot)$ are feature vectors of size κ_2 and κ_3, respectively. Although our estimates of $\rho(\theta)$ and $\eta(\theta)$ are unbiased, since we use biased estimates for V^θ and U^θ (linear approximations in the critic), our gradient estimates $\nabla\rho(\theta)$ and $\nabla\eta(\theta)$, and as a result $\nabla L(\theta,\lambda)$, are biased. Lemma 7 in Appendix C.2 shows the bias in our estimate of $\nabla L(\theta,\lambda)$. We prove that our actor-critic algorithm converges to a (local) saddle point of the risk-sensitive objective function $L(\theta,\lambda)$ (see Appendix C.3 in the supplementary material).

7 Experimental Results

We evaluate our algorithms in the context of a traffic signal control application. The objective in our formulation is to minimize the total number of vehicles in the system, which indirectly minimizes the delay experienced by the system. The motivation behind using a risk-sensitive control strategy is to reduce the variations in the delay experienced by road users.

We consider both infinite horizon discounted as well average settings for the traffic signal control MDP, formulated as in [14]. We briefly recall their formulation here: The state at

Algorithm 1 Template of the Average Reward Risk-Sensitive Actor-Critic Algorithm

Input: parameterized policy $\mu(\cdot|\cdot;\theta)$ and value function feature vectors $\phi_v(\cdot)$ and $\phi_u(\cdot)$
Initialization: policy parameters $\theta = \theta_0$; value function weight vectors $v = v_0$ and $u = u_0$; initial state $x_0 \sim P_0(x)$
for $t = 0, 1, 2, \ldots$ **do**
Draw action $a_t \sim \mu(\cdot|x_t;\theta_t)$
Observe next state $x_{t+1} \sim P(\cdot|x_t, a_t)$
Observe reward $R(x_t, a_t)$

Average Updates: $\widehat{\rho}_{t+1} = \big(1-\zeta_4(t)\big)\widehat{\rho}_t + \zeta_4(t)R(x_t,a_t), \quad \widehat{\eta}_{t+1} = \big(1-\zeta_4(t)\big)\widehat{\eta}_t + \zeta_4(t)R(x_t,a_t)^2$

TD Errors: $\delta_t = R(x_t,a_t) - \widehat{\rho}_{t+1} + v_t^\top \phi_v(x_{t+1}) - v_t^\top \phi_v(x_t)$

$\epsilon_t = R(x_t,a_t)^2 - \widehat{\eta}_{t+1} + u_t^\top \phi_u(x_{t+1}) - u_t^\top \phi_u(x_t)$

Critic Update: $v_{t+1} = v_t + \zeta_3(t)\delta_t\phi_v(x_t), \qquad u_{t+1} = u_t + \zeta_3(t)\epsilon_t\phi_u(x_t)$ (16)

Actor Update: $\theta_{t+1} = \Gamma\Big(\theta_t - \zeta_2(t)\big(-\delta_t\psi_t + \lambda_t(\epsilon_t\psi_t - 2\widehat{\rho}_{t+1}\delta_t\psi_t)\big)\Big)$ (17)

$\lambda_{t+1} = \Gamma_\lambda\Big(\lambda_t + \zeta_1(t)(\widehat{\eta}_{t+1} - \widehat{\rho}_{t+1}^2 - \alpha)\Big)$ (18)

end for
return policy and value function parameters θ, λ, v, u

each time t, x_t, is the vector of queue lengths and elapsed times and is given by $x_t = (q_1(t), \ldots, q_N(t), t_1(t), \ldots, t_N(t))$. Here q_i and t_i denote the queue length and elapsed time since the signal turned to red on lane i. The actions a_t belong to the set of feasible sign configurations. The single-stage cost function $h(x_t)$ is defined as follows:

$$h(x_t) = r_1\,\big[\sum_{i\in I_p} r_2\cdot q_i(t) + \sum_{i\notin I_p} s_2\cdot q_i(t)\big] + s_1\,\big[\sum_{i\in I_p} r_2\cdot t_i(t) + \sum_{i\notin I_p} s_2\cdot t_i(t)\big], \tag{19}$$

where $r_i, s_i \geq 0$ such that $r_i + s_i = 1$ for $i = 1, 2$ and $r_2 > s_2$. The set I_p is the set of prioritized lanes in the road network considered. While the weights r_1, s_1 are used to differentiate between the queue length and elapsed time factors, the weights r_2, s_2 help in prioritization of traffic.

Given the above traffic control setting, we aim to minimize both the long run discounted as well average sum of the cost function $h(x_t)$. The underlying policy for all the algorithms is a parameterized Boltzmann policy (see Appx. F). We implement the following algorithms in the discounted setting:
(i) Risk-neutral SPSA and SF algorithms with the actor update as follows:

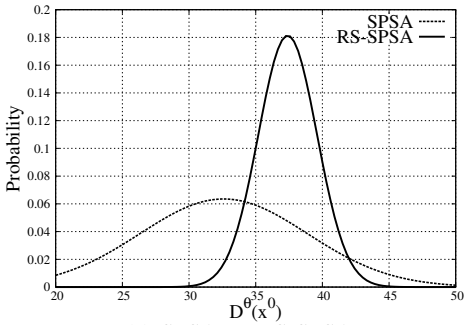
$$\theta_{t+1}^{(i)} = \Gamma_i\left(\theta_t^{(i)} + \frac{\zeta_2(t)}{\beta\Delta_t^{(i)}}(v_t^+ - v_t)^\top\phi_v(x^0)\right) \quad \textbf{SPSA},$$

$$\theta_{t+1}^{(i)} = \Gamma_i\left(\theta_t^{(i)} + \frac{\zeta_2(t)\Delta_t^{(i)}}{\beta}(v_t^+ - v_t)^\top\phi_v(x^0)\right) \quad \textbf{SF},$$

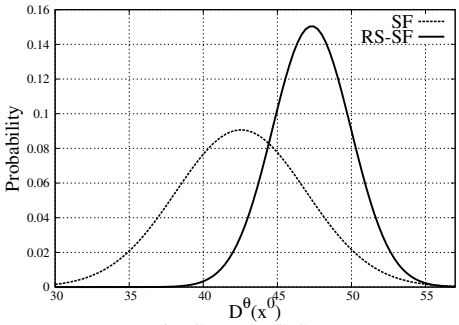
where the critic parameters v_t^+, v_t are updated according to (6). Note that these are two-timescale algorithms with a TD critic on the faster timescale and the actor on the slower timescale.
(ii) Risk-sensitive SPSA and SF algorithms (RS-SPSA and RS-SF) of Section 4 that attempt to solve (2) and update the policy parameter according to (8) and (9), respectively. In the average setting, we implement **(i)** the risk-neutral AC algorithm from [16] that incorporates an actor-critic scheme, and **(ii)** the risk-sensitive algorithm of Section 6 (RS-AC) that attempts to solve (12) and updates the policy parameter according to (17).

All our algorithms incorporate function approximation owing to the curse of dimensionality associated with larger road networks. For instance, assuming only 20 vehicles per lane of a 2x2-grid network, the cardinality of the state space is approximately of the order 10^{32} and the situation is aggravated as the size of the road network increases. The choice of features used in each of our algorithms is as described in Section V-B of [15]. We perform the experiments on a 2x2-grid network. The detailed list of parameters and step-sizes chosen for our algorithms is given in Appendix F.

Figures 2(a) and 2(b) show the distribution of the discounted cumulative reward $D^\theta(x^0)$ for the SPSA and SF algorithms, respectively. Figure 3(a) shows the distribution of the average reward ρ for the algorithms in the average setting. From these plots, we notice that the risk-sensitive algorithms

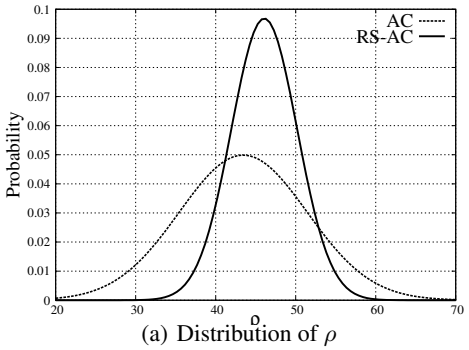


(a) SPSA vs. RS-SPSA

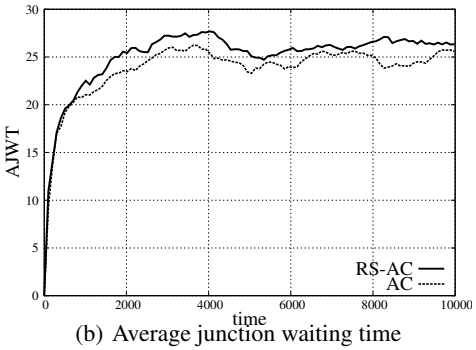


(b) SF vs. RS-SF

Figure 2: Performance comparison in the discounted setting using the distribution of $D^\theta(x^0)$.



(a) Distribution of ρ



(b) Average junction waiting time

Figure 3: Comparison of AC vs. RS-AC in the average setting using two different metrics.

that we propose result in a long-term (discounted or average) reward that is higher than their risk-neutral variants. However, from the empirical variance of the reward (both discounted as well as average) perspective, the risk-sensitive algorithms outperform their risk-neutral variants.

We use average junction waiting time (AJWT) to compare the algorithms from a traffic signal control application standpoint. Figure 3(b) presents the AJWT plots for the algorithms in the average setting (see Appendix F for similar results for the SPSA and SF algorithms in the discounted setting). We observe that the performance of our risk-sensitive algorithms is not significantly worse than their risk-neutral counterparts. This coupled with the observation that our algorithms exhibit low variance, makes them a suitable choice in risk-constrained systems.

8 Conclusions and Future Work

We proposed novel actor critic algorithms for control in risk-sensitive discounted and average reward MDPs. All our algorithms involve a TD critic on the fast timescale, a policy gradient (actor) on the intermediate timescale, and dual ascent for Lagrange multipliers on the slowest timescale. In the discounted setting, we pointed out the difficultly in estimating the gradient of the variance of the return and incorporated simultaneous perturbation based SPSA and SF approaches for gradient estimation in our algorithms. The average setting, on the other hand, allowed for an actor to employ compatible features to estimate the gradient of the variance. We provided proofs of convergence (in the appendix) to locally (risk-sensitive) optimal policies for all the proposed algorithms. Further, using a traffic signal control application, we observed that our algorithms resulted in lower variance empirically as compared to their risk-neutral counterparts.

In this paper, we established asymptotic limits for our discounted and average reward risk-sensitive actor-critic algorithms. To the best of our knowledge, there are no convergence rate results available for multi-timescale stochastic approximation schemes and hence for actor-critic algorithms. This is true even for the actor-critic algorithms that do not incorporate any risk criterion. It would be an interesting research direction to obtain finite-time bounds on the quality of the solution obtained by these algorithms.

References

[1] D. Bertsekas. *Nonlinear programming*. Athena Scientific, 1999.

[2] S. Bhatnagar, R. Sutton, M. Ghavamzadeh, and M. Lee. Natural actor-critic algorithms. *Automatica*, 45 (11):2471–2482, 2009.

[3] S. Bhatnagar, H. Prasad, and L. Prashanth. *Stochastic Recursive Algorithms for Optimization*, volume 434. Springer, 2013.

[4] V. Borkar. A sensitivity formula for the risk-sensitive cost and the actor-critic algorithm. *Systems & Control Letters*, 44:339–346, 2001.

[5] V. Borkar. Q-learning for risk-sensitive control. *Mathematics of Operations Research*, 27:294–311, 2002.

[6] H. Chen, T. Duncan, and B. Pasik-Duncan. A Kiefer-Wolfowitz algorithm with randomized differences. *IEEE Transactions on Automatic Control*, 44(3):442–453, 1999.

[7] E. Delage and S. Mannor. Percentile optimization for Markov decision processes with parameter uncertainty. *Operations Research*, 58(1):203–213, 2010.

[8] J. Filar, L. Kallenberg, and H. Lee. Variance-penalized Markov decision processes. *Mathematics of Operations Research*, 14(1):147–161, 1989.

[9] J. Filar, D. Krass, and K. Ross. Percentile performance criteria for limiting average Markov decision processes. *IEEE Transaction of Automatic Control*, 40(1):2–10, 1995.

[10] R. Howard and J. Matheson. Risk sensitive Markov decision processes. *Management Science*, 18(7): 356–369, 1972.

[11] V. Katkovnik and Y. Kulchitsky. Convergence of a class of random search algorithms. *Automatic Remote Control*, 8:81–87, 1972.

[12] A. Nilim and L. El Ghaoui. Robust control of Markov decision processes with uncertain transition matrices. *Operations Research*, 53(5):780–798, 2005.

[13] J. Peters, S. Vijayakumar, and S. Schaal. Natural actor-critic. In *Proceedings of the Sixteenth European Conference on Machine Learning*, pages 280–291, 2005.

[14] L.A. Prashanth and S. Bhatnagar. Reinforcement Learning With Function Approximation for Traffic Signal Control. *IEEE Transactions on Intelligent Transportation Systems*, 12(2):412 –421, june 2011.

[15] L.A. Prashanth and S. Bhatnagar. Threshold Tuning Using Stochastic Optimization for Graded Signal Control. *IEEE Transactions on Vehicular Technology*, 61(9):3865 –3880, nov. 2012.

[16] L.A. Prashanth and Shalabh Bhatnagar. Reinforcement learning with average cost for adaptive control of traffic lights at intersections. In *14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 1640–1645. IEEE, 2011.

[17] W. Sharpe. Mutual fund performance. *Journal of Business*, 39(1):119–138, 1966.

[18] M. Sobel. The variance of discounted Markov decision processes. *Applied Probability*, pages 794–802, 1982.

[19] J. Spall. Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE Transactions on Automatic Control*, 37(3):332–341, 1992.

[20] R. Sutton, D. McAllester, S. Singh, and Y. Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Proceedings of Advances in Neural Information Processing Systems 12*, pages 1057–1063, 2000.

[21] A. Tamar, D. Di Castro, and S. Mannor. Policy gradients with variance related risk criteria. In *Proceedings of the 29th International Conference on Machine Learning*, pages 387–396, 2012.

[22] A. Tamar, D. Di Castro, and S. Mannor. Temporal difference methods for the variance of the reward to go. In *Proceedings of the 30th International Conference on Machine Learning*, 2013.

[23] H. Xu and S. Mannor. Distributionally robust Markov decision processes. *Mathematics of Operations Research*, 37(2):288–300, 2012.

Appendix

A Assumptions

We make the following assumptions in the analysis of our discounted and average reward algorithms. Recall that Φ_v and Φ_u are $n \times \kappa_2$ and $n \times \kappa_3$ dimensional matrices (n is the total number of states in the state space $\mathcal{X}$) whose ith columns are $\phi_v^{(i)} = \big(\phi_v^{(i)}(x),\ x \in \mathcal{X}\big)^\top,\ i = 1, \ldots, \kappa_2$ and $\phi_u^{(i)} = \big(\phi_u^{(i)}(x),\ x \in \mathcal{X}\big)^\top,\ i = 1, \ldots, \kappa_3$.

(A1) *For any state-action pair (x, a), $\mu(a|x;\theta)$ is continuously differentiable in the parameter θ.*

(A2) *The Markov chain induced by any policy θ is irreducible and aperiodic.*

(A3) *The basis functions $\{\phi_v^{(i)}\}_{i=1}^{\kappa_2}$ and $\{\phi_u^{(i)}\}_{i=1}^{\kappa_3}$ are linearly independent. In particular, $\kappa_2, \kappa_3 \ll n$ and Φ_v and Φ_u are full rank. Moreover, for every $v \in \mathbb{R}^{\kappa_2}$ and $u \in \mathbb{R}^{\kappa_3}$, $\Phi_v v \neq e$ and $\Phi_u u \neq e$, where e is the n-dimensional vector with all entries equal to one.*

(A4) *The step size schedules $\{\zeta_4(t)\}$, $\{\zeta_3(t)\}$, $\{\zeta_2(t)\}$, and $\{\zeta_1(t)\}$ satisfy (k is some positive constant)*

$$\sum_t \zeta_1(t) = \sum_t \zeta_2(t) = \sum_t \zeta_3(t) = \infty, \tag{1}$$

$$\sum_t \zeta_1(t)^2, \quad \sum_t \zeta_2(t)^2, \quad \sum_t \zeta_3(t)^2 < \infty, \tag{2}$$

$$\zeta_1(t) = o\big(\zeta_2(t)\big), \quad \zeta_2(t) = o\big(\zeta_3(t)\big), \quad \zeta_4(t) = k\zeta_3(t). \tag{3}$$

Equations 1 and 2 are standard step-size conditions in stochastic approximation algorithms, and Equation 3 indicates that the updates correspond to $\{\zeta_3(t)\}$ and $\{\zeta_4(t)\}$ are on the (same) fastest time-scale, the update corresponds to $\{\zeta_2(t)\}$ is on the intermediate time-scale, and the update corresponds to $\{\zeta_1(t)\}$ is on the slowest time-scale.

B Discounted Reward Setting, Algorithms, and Analysis

B.1 Bellman Equations for Square Value and Action-value Functions

$$U^\mu(x) = \sum_a \mu(a|x) r(x,a)^2 + \gamma^2 \sum_{a,x'} \mu(a|x) P(x'|x,a) U^\mu(x') + 2\gamma \sum_{a,x'} \mu(a|x) P(x'|x,a) r(x,a) V^\mu(x'),$$
$$W^\mu(x,a) = r(x,a)^2 + \gamma^2 \sum_{x'} P(x'|x,a) U^\mu(x') + 2\gamma r(x,a) \sum_{x'} P(x'|x,a) V^\mu(x'). \tag{4}$$

B.2 Gradient of the Risk-sensitive Criterion

Lemma 1 *Under Assumption (A1), we have*

$$(1-\gamma)\nabla V^\theta(x^0) = \sum_{x,a} \pi^\theta_\gamma(x,a|x^0) \nabla \log \mu(a|x;\theta) Q^\theta(x,a),$$
$$(1-\gamma^2)\nabla U^\theta(x^0) = \sum_{x,a} \widetilde{\pi}^\theta_\gamma(x,a|x^0) \nabla \log \mu(a|x;\theta) W^\theta(x,a) + 2\gamma \sum_{x,a,x'} \widetilde{\pi}^\theta_\gamma(x,a|x^0) P(x'|x,a) r(x,a) \nabla V^\theta(x'),$$

where $\widetilde{\pi}^\theta_\gamma(x,a|x^0) = \widetilde{d}^\theta_\gamma(x|x^0)\mu(a|x)$ *and* $\widetilde{d}^\theta_\gamma(x|x^0) = (1-\gamma^2)\sum_{t=0}^\infty \gamma^{2t} \Pr(x_t = x | x_0 = x^0; \theta)$.

Proof. The proof of $\nabla V^\theta(x^0)$ can be found in the literature (e.g., [9]). To prove $\nabla U^\theta(x^0)$, we start by the fact that from (4) we have $U(x) = \sum_a \mu(x|a) W(x,a)$. If we take the derivative w.r.t. θ from both sides of this equation, we obtain

$$\begin{aligned}
\nabla U(x^0) =& \sum_a \nabla \mu(x^0|a) W(x^0,a) + \sum_a \mu(a|x^0) \nabla W(x^0,a) \\
=& \sum_a \nabla \mu(a|x^0) W(x^0,a) + \sum_a \mu(a|x^0) \nabla \Big[r(x^0,a)^2 + \gamma^2 \sum_{x'} P(x'|x^0,a) U(x') \\
& + 2\gamma r(x^0,a) \sum_{x'} P(x'|x^0,a) V(x') \Big] \\
=& \underbrace{\sum_a \nabla \mu(x^0|a) W(x^0,a) + 2\gamma \sum_{a,x'} \mu(a|x^0) r(x^0,a) P(x'|x^0,a) \nabla V(x')}_{h(x^0)} \\
& + \gamma^2 \sum_{a,x'} \mu(a|x^0) P(x'|x^0,a) \nabla U(x') \\
=& h(x^0) + \gamma^2 \sum_{a,x'} \mu(a|x^0) P(x'|x^0,a) \nabla U(x') \qquad (5) \\
=& h(x^0) + \gamma^2 \sum_{a,x'} \mu(a|x^0) P(x'|x^0,a) \nabla \Big[h(x') + \gamma^2 \sum_{a',x''} \mu(a'|x') P(x''|x',a') \nabla U(x'') \Big]
\end{aligned}$$

By unrolling the last equation using the definition of $\nabla U(x)$ from (5), we obtain

$$\begin{aligned}
\nabla U(x^0) &= \sum_{t=0}^\infty \gamma^{2t} \sum_x \Pr(x_t = x | x_0 = x^0) h(x) = \frac{1}{1-\gamma^2} \sum_x \widetilde{d}_\gamma(x|x^0) h(x) \\
&= \frac{1}{1-\gamma^2} \Big[\sum_{x,a} \widetilde{d}_\gamma(x|x^0) \mu(a|x) \nabla \log \mu(a|x) W(x,a) + 2\gamma \sum_{x,a,x'} \widetilde{d}_\gamma(x|x^0) \mu(a|x) r(x,a) P(x'|x,a) \nabla V(x') \Big] \\
&= \frac{1}{1-\gamma^2} \Big[\sum_{x,a} \widetilde{\pi}_\gamma(x,a|x^0) \nabla \log \mu(a|x) W(x,a) + 2\gamma \sum_{x,a,x'} \widetilde{\pi}_\gamma(x,a|x^0) r(x,a) P(x'|x,a) \nabla V(x') \Big].
\end{aligned}$$

■

B.3 Convergence Analysis of the Risk-Sensitive SPSA and SF Actor-Critic Algorithms

Our proposed actor-critic algorithms use multi-timescale stochastic approximation and we use the ordinary differential equation (ODE) approach (see Chapter 6 of [4]) to analyze their convergence. The proof of convergence of the SPSA and SF algorithms to a (local) saddle point of the risk-sensitive objective function $\widehat{L}(\theta,\lambda) \stackrel{\triangle}{=} -\widehat{V}^{\theta}(x^0) + \lambda(\widehat{\Lambda}^{\theta}(x^0) - \alpha) = -\widehat{V}^{\theta}(x^0) + \lambda\big(\widehat{U}^{\theta}(x^0) - \widehat{V}^{\theta}(x^0)^2 - \alpha\big)$ contains the following three main steps. Note that since SPSA and SF use different methods to estimate the gradient, their proofs only differ in the second step, i.e., the convergence of the policy parameter θ.

As mentioned above, we establish asymptotic limits for both our algorithms using the ODE approach. To the best of our knowledge, there are no convergence rate results available for multi-timescale stochastic approximation schemes and hence for actor-critic algorithms. This is true even for the actor-critic algorithms that do not incorporate any risk criterion. It would be an interesting orthogonal direction of research to obtain finite-time bounds on the quality of the solution obtained by these algorithms.

Note that in the following analysis, we use the assumptions (A1) to (A4) defined in Appendix A.

Step 1: (Critic's Convergence)

The goal here is to show that the value and square value estimates of policies θ and $\theta^+ = \theta + \beta\Delta$ converge. Since the critic's update is on the fastest time-scale and the step-size schedules satisfy (A4), we can assume in this analysis that θ and λ are time invariant quantities.

Theorem 2 *Under (A1)-(A4), for any given policy parameter θ and Lagrange multiplier λ, the critic parameters $\{v_t\},\{v_t^+\}$ and $\{u_t\},\{u_t^+\}$ governed by recursions of Eq. 6 in the paper, converges, i.e., $v_t \to \bar{v}, v_t^+ \to \bar{v}^+$ and $u_t \to \bar{u}, u_t^+ \to \bar{u}^+$, where $\bar{v}, \bar{v}^+$ and $\bar{u}, \bar{u}^+$ are the unique solutions to*

$$(\Phi_v^\top \boldsymbol{D}_\gamma^\theta \Phi_v)\bar{v} = \Phi_v^\top \boldsymbol{D}_\gamma^\theta T_v^\theta[\Phi_v\bar{v}], \qquad (\Phi_v^\top \boldsymbol{D}_\gamma^{\theta^+} \Phi_v)\bar{v}^+ = \Phi_v^\top \boldsymbol{D}_\gamma^{\theta^+} T_v^{\theta^+}[\Phi_v\bar{v}^+],$$
$$(\Phi_v^\top \boldsymbol{D}_\gamma^\theta \Phi_u)\bar{u} = \Phi_v^\top \boldsymbol{D}_\gamma^\theta T_u^\theta[\Phi_u\bar{u}], \qquad (\Phi_v^\top \boldsymbol{D}_\gamma^{\theta^+} \Phi_u)\bar{u}^+ = \Phi_v^\top \boldsymbol{D}_\gamma^{\theta^+} T_u^{\theta^+}[\Phi_u\bar{u}^+],$$

where n is the total number states in the state space $\mathcal{X}$, and

- *Φ_v and Φ_u are $n \times \kappa_2$ and $n \times \kappa_3$ dimensional matrices $(\kappa_2, \kappa_3 \ll n)$ whose i'th columns are $\phi_v^{(i)} = \big(\phi_v^{(i)}(x),\ x \in \mathcal{X}\big)^\top,\ i = 1,\ldots,\kappa_2$ and $\phi_u^{(i)} = \big(\phi_u^{(i)}(x),\ x \in \mathcal{X}\big)^\top,\ i = 1,\ldots,\kappa_3$.*
- *$\boldsymbol{D}_\gamma^\theta$ and $\boldsymbol{D}_\gamma^{\theta^+}$ denote the diagonal matrices with entries $d_\gamma^\theta(x)$ and $d_\gamma^{\theta^+}(x)$ for all $x \in \mathcal{X}$.*
- *$T_v^\theta, T_v^{\theta^+}$ and $T_u^\theta, T_u^{\theta^+}$ are the Bellman operators for value and square value functions of policies θ and θ^+, respectively. For any $y \in \mathbb{R}^{2n}$ such that $y = [y_v; y_u]$ and $y_v, y_u \in \mathbb{R}^n$, these operators are defined as $T_v^\theta y = \boldsymbol{r}^\theta + \gamma \boldsymbol{P}^\theta y_v$ and $T_u^\theta y = \boldsymbol{R}^\theta \boldsymbol{r}^\theta + 2\gamma \boldsymbol{R}^\theta \boldsymbol{P}^\theta y_v + \gamma^2 \boldsymbol{P}^\theta y_u$, where $\boldsymbol{r}^\theta$ and $\boldsymbol{P}^\theta$ are the reward vector and transition probability matrix of policy θ, and $\boldsymbol{R}^\theta = diag(\boldsymbol{r}^\theta)$.*

Proof. The proof of this theorem follows similar steps as in the proof of Theorem 10 in Tamar et al. [14]. For our analysis, we need to extend their proof to discounted MDPs and to the case that the reward is a function of both states and actions (and not just states), which is straightforward. ■

Remark 1 *Note that $[\Phi_v\bar{v}; \Phi_u\bar{u}]$ (the value and square value functions that the critic converges to) is the unique fixed point of the projected Bellman operator ΠT, where T contains both Bellman operators T_v and T_u for value and square value functions and Π contains both projections Π_v and Π_u into the linear spaces spanned by the columns of Φ_v and Φ_u (see [14] for more details).*

Step 2: (Analysis of θ-recursion)

Here we show that the update of θ is equivalent to gradient descent for the function $\widehat{L}(\theta,\lambda) \stackrel{\triangle}{=} -\widehat{V}^{\theta}(x^0) + \lambda(\widehat{\Lambda}^{\theta}(x^0) - \alpha) = -\widehat{V}^{\theta}(x^0) + \lambda\big(\widehat{U}^{\theta}(x^0) - \widehat{V}^{\theta}(x^0)^2 - \alpha\big)$ and converges to a limiting set that depends on λ. Consider the ODE

$$\dot{\theta}_t = \check{\Gamma}\left(\nabla_\theta \widehat{L}(\theta_t,\lambda)\right), \tag{6}$$

where $\check{\Gamma}$ is defined as follows: For any bounded continuous function $f(\cdot)$,

$$\check{\Gamma}\big(f(\theta_t)\big) = \lim_{\tau\to 0}\frac{\Gamma\big(\theta_t + \tau f(\theta_t)\big) - \theta_t}{\tau}. \tag{7}$$

The projection operator $\check{\Gamma}(\cdot)$ ensures that the evolution of θ via the ODE (6) stays within the bounded set $C \in \mathbb{R}^{\kappa_1}$. Due to timescale separation, the value of λ (updated on a slower timescale) is assumed to be constant for the analysis of the θ-update.

Let $\mathcal{Z}_\lambda = \big\{\theta \in C : \check{\Gamma}\big(\nabla \widehat{L}(\theta_t,\lambda)\big) = 0\big\}$ denote the set of asymptotically stable equilibrium points of the ODE (6) and $\mathcal{Z}_\lambda^\varepsilon = \big\{\theta \in C : ||\theta - \theta_0|| < \varepsilon, \theta_0 \in \mathcal{Z}_\lambda\big\}$ denote the set of points in the ε-neighborhood of $\mathcal{Z}_\lambda$. The main result regarding the convergence of the policy parameter θ for both the SPSA and SF algorithms is as follows:

Theorem 3 *Under (A1)-(A4), for any given Lagrange multiplier λ and $\varepsilon > 0$, there exists $\beta_0 > 0$ such that for all $\beta \in (0,\beta_0)$, $\theta_t \to \theta^* \in \mathcal{Z}_\lambda^\varepsilon$ almost surely.*

Proof. **(Theorem 3 for SPSA)** Since the TD critic converges on the faster timescale, the θ-update in Eq. 8 of the paper can be rewritten using the converged TD-parameters $(\bar{v},\bar{u})$ and $(\bar{v}^+,\bar{u}^+)$ as

$$\theta_{t+1}^{(i)} = \Gamma_i\bigg(\theta_t^{(i)} - \zeta_2(t)\Big(-\big(1 + 2\lambda\bar{v}^\top\phi_v(x^0)\big)\frac{(\bar{v}^+ - \bar{v})^\top\phi_v(x^0)}{\beta\Delta_t^{(i)}} + \lambda\frac{(\bar{u}^+ - \bar{u})^\top\phi_u(x^0)}{\beta\Delta_t^{(i)}} + \xi_{1,t}\Big)\bigg),$$

where $\xi_{1,t} \to 0$ (convergence of TD in the critic and as a result convergence of the critic's parameters to $\bar{v},\bar{u},\bar{v}^+,\bar{u}^+$) in lieu of Theorem 2.

Next, we establish that $\mathbb{E}\left[\frac{(\bar{v}^+ - \bar{v})^\top\phi_v(x^0)}{\beta\Delta^{(i)}} \mid \theta,\lambda\right]$ is a biased estimator of $\nabla_\theta\widehat{V}(\theta)$, where the bias vanishes asymptotically.

$$\mathbb{E}\left[\frac{(\bar{v}^+ - \bar{v})^\top\phi_v(x^0)}{\beta\Delta^{(i)}} \mid \theta,\lambda\right] = \partial_{\theta^{(i)}}\bar{v}^\top\phi_v(x^0) + \mathbb{E}\left[\sum_{j\neq i}\frac{\Delta^{(j)}}{\Delta^{(i)}}\partial_{\theta^{(j)}}\bar{v}^\top\phi_v(x^0) \mid \theta,\lambda\right] + \xi_{2,t}\phi_v(x^0)$$
$$\longrightarrow_{\beta\to 0} \partial_{\theta^{(i)}}\bar{v}^\top\phi_v(x^0).$$

The first equality above follows by expanding using Taylor series, whereas the second step follows by using the fact that $\Delta_t^{(i)}$'s are independent Rademacher random variables. On similar lines, it can be seen that

$$\mathbb{E}\left[\frac{(\bar{u}^+ - \bar{u})^\top\phi_u(x^0)}{\beta\Delta^{(i)}} \mid \theta,\lambda\right] \longrightarrow_{\beta\to 0} \partial_{\theta^{(i)}}\bar{u}^\top\phi_u(x^0).$$

Thus, Eq. 8 in the paper can be seen to be a discretization of the ODE (6). Further, $\mathcal{Z}_\lambda$ is an asymptotically stable attractor for the ODE (6), with $\widehat{L}(\theta,\lambda)$ itself serving as a strict Lyapunov function. This can be inferred as follows:

$$\frac{d\widehat{L}(\theta,\lambda)}{dt} = \nabla_\theta\widehat{L}(\theta,\lambda)\dot{\theta} = \nabla_\theta\widehat{L}(\theta,\lambda)\check{\Gamma}\big(-\nabla_\theta\widehat{L}(\theta,\lambda)\big) < 0.$$

The claim now follows from Theorem 5.3.3, pp. 191-196 of Kushner and Clark [7]. ■

Proof. **(Theorem 3 for SF)** As in the case of the SPSA algorithm, we rewrite the θ-update in Eq. 9 of the paper using the converged TD-parameters as

$$\theta_{t+1}^{(i)} = \Gamma_i\Big(\theta_t^{(i)} - \zeta_2(t)\Big(\frac{-\Delta_t^{(i)}\big(1+2\lambda\bar{v}^\top\phi_v(x^0)\big)}{\beta}(\bar{v}^+ - \bar{v})^\top\phi_v(x^0) + \frac{\lambda\Delta_t^{(i)}}{\beta}(\bar{u}^+ - \bar{u})^\top\phi_u(x^0) + \xi_{1,t}\Big)\Big),$$

where $\xi_{1,t} \to 0$ (convergence of TD in the critic and as a result convergence of the critic's parameters to $\bar{v}, \bar{u}, \bar{v}^+, \bar{u}^+$) in lieu of Theorem 2. Next, we establish that $\mathbb{E}\left[\frac{\Delta^{(i)}}{\beta}(\bar{v}^+ - \bar{v})^\top\phi_v(x^0) \mid \theta, \lambda\right]$ is an asymptotically correct estimate of the gradient of $\widehat{V}(\theta)$ in the following:

$$\mathbb{E}\left[\frac{\Delta^{(i)}}{\beta}(\bar{v}^+ - \bar{v})^\top\phi_v(x^0) \mid \theta, \lambda\right] \longrightarrow_{\beta\to 0} \partial_{\theta^{(i)}}\bar{v}^\top\phi_v(x^0).$$

The above follows in a similar manner as Proposition 10.2 of Bhatnagar et al. [3]. On similar lines, one can see that

$$\mathbb{E}\left[\frac{\Delta^{(i)}}{\beta}(\bar{u}^+ - \bar{u})^\top\phi_u(x^0) \mid \theta, \lambda\right] \longrightarrow_{\beta\to 0} \partial_{\theta^{(i)}}\bar{u}^\top\phi_u(x^0).$$

Thus, Eq. 9 in the paper can be seen to be a discretization of the ODE (6) and the rest of the analysis follows in a similar manner as in the SPSA proof. ■

Step 3: (Analysis of λ-recursion and Convergence to a Local Saddle Point)

The goal here is to first show that the λ-recursion converges and then to prove that the whole algorithm converges to a local saddle point of $\widehat{L}(\theta, \lambda)$. We define the following ODE governing the evolution of λ

$$\dot{\lambda}_t \;=\; \check{\Gamma}_\lambda\big[\widehat{\Lambda}^{\theta_t}(x^0) - \alpha\big] \;=\; \check{\Gamma}_\lambda\big[\widehat{U}^{\theta_t}(x^0) - \widehat{V}^{\theta_t}(x^0)^2 - \alpha\big]. \tag{8}$$

where $\check{\Gamma}_\lambda$ is defined as follows: For any bounded continuous function $f(\cdot)$,

$$\check{\Gamma}_\lambda\big(f(\lambda_t)\big) = \lim_{\tau\to 0}\frac{\Gamma\big(\lambda_t + \tau f(\lambda_t)\big) - \lambda_t}{\tau}. \tag{9}$$

The operator $\check{\Gamma}_\lambda$ is similar to the operator $\check{\Gamma}$ defined in (7).

Theorem 4 *$\lambda_t \to \mathcal{F}$ almost surely as $t \to \infty$, where $\mathcal{F} \triangleq \{\lambda \mid \lambda \in [0, \lambda_{\max}],\ \check{\Gamma}_\lambda\big[\widehat{\Lambda}^{\theta^\lambda}(x^0) - \alpha\big] = 0,\ \theta^\lambda \in \mathcal{Z}_\lambda\}$.*

The last step is to establish that the algorithm converges to a (local) saddle point of $\widehat{L}(\theta, \lambda)$. In other words, to a pair (θ^*, λ^*) that are a local minimum w.r.t. θ and a local maximum w.r.t. λ of $\widehat{L}(\theta, \lambda)$. From Theorem 4, $\lambda_t \to \lambda^*$ for some $\lambda^* \in [0, \lambda_{\max}]$ such that $\theta^{\lambda^*} \in Z_{\lambda^*}$ and $\check{\Gamma}_\lambda\big[\widehat{\Lambda}^{\theta^{\lambda^*}}(x^0) - \alpha\big] = 0$. We now invoke the envelope theorem of mathematical economics [8] to conclude that the ODE $\dot{\lambda}_t = \check{\Gamma}_\lambda\big[\widehat{\Lambda}^{\theta_t}(x^0) - \alpha\big]$ is equivalent to $\dot{\lambda}_t = \check{\Gamma}_\lambda\big[\nabla_\lambda\widehat{L}(\theta^{\lambda^*}, \lambda^*)\big]$. From the above, it is clear that (θ_t, λ_t) governed by Eqs. 8 to 10 in the paper converges to a local saddle point of $\widehat{L}(\theta, \lambda)$.

Proof. The proof follows the same steps as in Theorem 3 in Bhatnagar [1]. ■

C Average Reward Setting, Algorithm, and Analysis

Note that in this section, we use the assumptions (A1) to (A4) defined in Appendix A.

C.1 Gradient of the Risk-sensitive Criterion

Lemma 5 *Under Assumptions (A1) and (A2), we have*

$$\nabla\rho(\theta) = \sum_{x,a} \pi(x,a;\theta)\nabla \log \mu(a|x;\theta) Q(x,a;\theta), \tag{10}$$

$$\nabla\eta(\theta) = \sum_{x,a} \pi(x,a;\theta)\nabla \log \mu(a|x;\theta) W(x,a;\theta). \tag{11}$$

Proof. **P**roof of $\nabla\rho(\theta)$ can be found in [12] and [6]. To prove $\nabla\eta(\theta)$, we start by the fact that from Eq. 15 in the paper, we have $U(x) = \sum_a \mu(x|a)W(x,a)$. If we take the derivative w.r.t. θ from both sides of this equation, we obtain

$$\begin{aligned}
\nabla U(x) &= \sum_a \nabla\mu(x|a)W(x,a) + \sum_a \mu(x|a)\nabla W(x,a) \\
&= \sum_a \nabla\mu(x|a)W(x,a) + \sum_a \mu(x|a)\nabla\big(r(x,a)^2 - \eta + \sum_{x'} P(x'|x,a)U(x')\big) \\
&= \sum_a \nabla\mu(x|a)W(x,a) - \nabla\eta + \sum_{a,x'} \mu(a|x)P(x'|x,a)\nabla U(x'). \qquad (12)
\end{aligned}$$

The second equality is by replacing $W(x,a)$ from Eq. 15 of the paper. Now if we take the weighted sum, weighted by $d(x)$, from both sides of (12), we have

$$\sum_x d(x)\nabla U(x) = \sum_{x,a} d(x)\nabla\mu(a|x)W(x,a) - \nabla\eta + \sum_{a,x'} d(x)\mu(a|x)P(x'|x,a)\nabla U(x'). \tag{13}$$

The claim follows from the fact that the last sum on the RHS of (13) is equal to $\sum_x d(x)\nabla U(x)$. ■

C.2 Bias in the Gradient Estimate

We first show that if $\widehat{V}$, $\widehat{U}$, $\widehat{\rho}$, and $\widehat{\eta}$ are unbiased estimators of V^μ, U^μ, $\rho(\mu)$, and $\eta(\mu)$, respectively, then δ_t and ϵ_t are unbiased estimates of the advantage functions A^μ and B^μ.

Lemma 6 *For any given policy μ, we have*

$$\mathbb{E}[\,\delta_t|\,x_t, a_t, \mu] = A^\mu(x_t, a_t), \qquad\qquad \mathbb{E}[\,\epsilon_t|\,x_t, a_t, \mu] = B^\mu(x_t, a_t).$$

Proof. The first statement $\mathbb{E}[\,\delta_t|\,x_t, a_t, \mu] = A^\mu(x_t, a_t)$ has been proved in Lemma 3 of [2], so here we only prove the second statement $\mathbb{E}[\,\epsilon_t|\,x_t, a_t, \mu] = B^\mu(x_t, a_t)$. we may write

$$\begin{aligned}
\mathbb{E}[\,\epsilon_t|\,x_t, a_t, \mu] &= \mathbb{E}\big[R(x_t, a_t)^2 - \widehat{\eta}_{t+1} + \widehat{U}(x_{t+1}) - \widehat{U}(x_t) \mid x_t, a_t, \mu\big] \\
&= r(x_t, a_t)^2 - \eta(\mu) + \mathbb{E}\big[\widehat{U}(x_{t+1}) \mid x_t, a_t, \mu\big] - U^\mu(x_t) \\
&= r(x_t, a_t)^2 - \eta(\mu) + \mathbb{E}\Big[\mathbb{E}\big[\widehat{U}(x_{t+1}) \mid x_{t+1}, \mu\big] \mid x_t, a_t\Big] - U^\mu(x_t) \\
&= r(x_t, a_t)^2 - \eta(\mu) + \mathbb{E}\big[\widehat{U}(x_{t+1}) \mid x_t, a_t\big] - U^\mu(x_t) \\
&= \underbrace{r(x_t, a_t)^2 - \eta(\mu) + \sum_{x_{t+1}\in\mathcal{X}} P(x_{t+1}|x_t, a_t) U^\mu(x_{t+1})}_{W^\mu(x,a)} - U^\mu(x_t) = B^\mu(x, a).
\end{aligned}$$

■

Although our estimates of $\rho(\theta)$ and $\eta(\theta)$ are unbiased, since we use biased estimates for V^θ and U^θ in the critic, our gradient estimates $\nabla\rho(\theta)$ and $\nabla\eta(\theta)$ are biased. As a result, our estimate of $\nabla L(\theta, \lambda)$ is also biased and the following lemma quantifies this bias.

Lemma 7 *The bias of our actor-critic algorithm in estimating $\nabla L(\theta, \lambda)$ for fixed θ and λ is*

$$\mathcal{B}(\theta, \lambda) = \sum_x d^\theta(x)\Big\{ -\big(1 + 2\lambda\rho(\theta)\big)\big[\nabla\bar{V}^\theta(x) - \nabla {v^\theta}^\top \phi_v(x)\big] + \lambda\big[\nabla\bar{U}^\theta(x) - \nabla {u^\theta}^\top \phi_u(x)\big]\Big\},$$

where ${v^\theta}^\top \phi_v(\cdot)$ and ${u^\theta}^\top \phi_u(\cdot)$ are estimates of $V^\theta(\cdot)$ and $U^\theta(\cdot)$ upon convergence of the TD recursion, and

$$\begin{aligned}
\bar{V}^\theta(x) &= \sum_a \mu(a|x)\big[r(x, a) - \rho(\theta) + \sum_{x'} P(x'|x, a) {v^\theta}^\top \phi_v(x')\big], \\
\bar{U}^\theta(x) &= \sum_a \mu(a|x)\big[r(x, a)^2 - \eta(\theta) + \sum_{x'} P(x'|x, a) {u^\theta}^\top \phi_u(x')\big].
\end{aligned}$$

Proof. The bias in estimating $\nabla L(\theta, \lambda)$ consists of the bias in estimating $\nabla\rho(\theta)$ and $\nabla\eta(\theta)$. Lemma 4 in Bhatnagar et al. [2] shows the bias in estimating $\nabla\rho(\theta)$ as

$$\mathbb{E}[\delta_t^\theta \psi_t|\theta] = \nabla\rho(\theta) + \sum_{x\in\mathcal{X}} d^\theta(x)\big[\nabla\bar{V}^\theta(x) - \nabla {v^\theta}^\top \phi_v(x)\big],$$

where $\delta_t^\theta = R(x_t, a_t) - \widehat{\rho}_{t+1} + {v^\theta}^\top \phi_v(x_{t+1}) - {v^\theta}^\top \phi_v(x_t)$. Similarly we can prove that the bias in estimating $\nabla\eta(\theta)$ is

$$\mathbb{E}[\epsilon_t^\theta \psi_t|\theta] = \nabla\eta(\theta) + \sum_{x\in\mathcal{X}} d^\theta(x)\big[\nabla\bar{U}^\theta(x) - \nabla {u^\theta}^\top \phi_u(x)\big],$$

where $\epsilon_t^\theta = R(x_t, a_t) - \widehat{\eta}_{t+1} + {u^\theta}^\top \phi_u(x_{t+1}) - {u^\theta}^\top \phi_u(x_t)$. The claim follows by putting these two results together and given the fact that $\nabla\Lambda(\theta) = \nabla\eta(\theta) - 2\rho(\theta)\nabla\rho(\theta)$ and $\nabla L(\theta, \lambda) = -\nabla\rho(\theta) + \lambda\nabla\Lambda(\theta)$. Note that the following fact holds for the bias in estimating $\nabla\rho(\theta)$ and $\nabla\eta(\theta)$.

$$\sum_x d^\theta(x)\big[\bar{V}^\theta(x) - {v^\theta}^\top \phi_v(x)\big] = 0, \qquad\qquad \sum_x d^\theta(x)\big[\bar{U}^\theta(x) - {u^\theta}^\top \phi_u(x)\big] = 0.$$

■

C.3 Convergence Analysis of the Average Reward Risk-Sensitive Actor-Critic Algorithm

As in the discounted setting in Appendix B.3, we use the ODE approach [4] to analyze the convergence of our average reward risk-sensitive actor-critic algorithm. The proof involves three main steps:

1. The first step is the convergence of ρ, η, V, and U, for any fixed policy θ and Lagrange multiplier λ. This corresponds to a TD(0) (with extension to η and U) proof. The policy and Lagrange multiplier are considered fixed because the critic's updates are on the faster time-scale than the actor's.
2. The second step is to show the convergence of θ_t to an ε-neighborhood $\mathcal{Z}_\lambda^\varepsilon$ of the set of asymptotically stable equilibria $\mathcal{Z}_\lambda$ of ODE

$$\dot{\theta}_t = \check{\Gamma}\big(\nabla L(\theta_t, \lambda)\big), \tag{14}$$

   where for any bounded continuous function $f(\cdot)$, the projection operator $\check{\Gamma}$ is defined as

$$\check{\Gamma}\big(f(\theta_t)\big) = \lim_{\tau \to 0} \frac{\Gamma\big(\theta_t + \tau f(\theta_t)\big) - \theta_t}{\tau}. \tag{15}$$

   $\check{\Gamma}$ ensures that the evolution of θ via the ODE (14) stays within the compact and convex set $C \subset \mathbb{R}^{\kappa_1}$. Again here it is assumed that λ is fixed because θ-recursion is on a faster time-scale than λ's.
3. The final step is the convergence of λ and showing that the whole algorithm converges to a local saddle point of $L(\theta, \lambda)$.

Step 1: Critic's Convergence

Lemma 8 *For any given policy μ, $\{\widehat{\rho}_t\}$, $\{\widehat{\eta}_t\}$, $\{v_t\}$, and $\{u_t\}$, defined in Algorithm 1 and by the critic recursion of Eq. 16 in the paper, converge to $\rho(\mu)$, $\eta(\mu)$, v^μ, and u^μ with probability one, where v^μ and u^μ are the unique solution to*

$$\Phi_v^\top \boldsymbol{D}^\mu \Phi_v v^\mu = \Phi_v^\top \boldsymbol{D}^\mu T_v^\mu(\Phi_v v^\mu), \qquad \Phi_u^\top \boldsymbol{D}^\mu \Phi_u u^\mu = \Phi_u^\top \boldsymbol{D}^\mu T_u^\mu(\Phi_u u^\mu), \tag{16}$$

respectively. In (16), $\boldsymbol{D}^\mu$ denotes the diagonal matrix with entries $d^\mu(x)$ for all $x \in \mathcal{X}$, and T_v^μ and T_u^μ are the Bellman operators for the differential value and square value functions of policy μ, defined as

$$T_v^\mu J = \boldsymbol{r}^\mu - \rho(\mu)\boldsymbol{e} + \boldsymbol{P}^\mu J, \qquad T_u^\mu J = \boldsymbol{R}^\mu \boldsymbol{r}^\mu - \eta(\mu)\boldsymbol{e} + \boldsymbol{P}^\mu J, \tag{17}$$

where $\boldsymbol{r}^\mu$ and $\boldsymbol{P}^\mu$ are reward vector and transition probability matrix of policy μ, $\boldsymbol{R}^\mu = diag(\boldsymbol{r}^\mu)$, and $\boldsymbol{e}$ is a vector of size n (the size of the state space $\mathcal{X}$) with elements all equal to one.

Proof. The proof follows the same steps as Lemma 5 in [2]. ■

Step 2: Actor's Convergence

Lemma 9 *Under Assumptions (A1)-(A4), given $\varepsilon > 0$, $\exists \delta > 0$ such that for θ_t, $t \geq 0$ obtained using Algorithm 1, if $\sup_\theta \|\mathcal{B}(\theta, \lambda)\| < \delta$ then $\theta_t \to \mathcal{Z}_\lambda^\varepsilon$ as $t \to \infty$ with probability one.*

Proof. First note that the bias of Algorithm 1 in estimating $\nabla L(\theta, \lambda)$ is (see Lemma 7 in Appendix C.2)

$$\mathcal{B}(\theta, \lambda) = \sum_x d^\theta(x)\Big\{ -\big(1 + 2\lambda\rho(\theta)\big)\big[\nabla \bar{V}^\theta(x) - \nabla v^{\theta\top}\phi_v(x)\big] + \lambda\big[\nabla \bar{U}^\theta(x) - \nabla u^{\theta\top}\phi_u(x)\big]\Big\}.$$

Also note that $\mathcal{Z}_\lambda = \big\{\theta \in C : \check{\Gamma}\big(-\nabla L(\theta, \lambda)\big) = 0\big\}$ denote the set of asymptotically stable equilibrium points of the ODE (6) and $\mathcal{Z}_\lambda^\varepsilon = \big\{\theta \in C : ||\theta - \theta_0|| < \varepsilon, \theta_0 \in \mathcal{Z}_\lambda\big\}$ denote the set of points in the ε-neighborhood of $\mathcal{Z}_\lambda$.

Let $\mathcal{F}(t) = \sigma(\theta_r, r \leq t)$ denote the sequence of σ-fields generated by $\theta_r,\ r \geq 0$. We have

$$\begin{aligned}
\theta_{t+1} &= \Gamma\Big(\theta_t - \zeta_2(t)\big(-\delta_t\psi_t + \lambda(\epsilon_t\psi_t - 2\widehat{\rho}_{t+1}\delta_t\psi_t)\big)\Big) \\
&= \Gamma\big(\theta_t + \zeta_2(t)(1 + 2\lambda\widehat{\rho}_{t+1})\delta_t\psi_t - \zeta_2(t)\lambda\epsilon_t\psi_t\big) \\
&= \Gamma\Bigg(\theta_t - \zeta_2(t)\Big[1 + 2\lambda\Big(\big(\widehat{\rho}_{t+1} - \rho(\theta_t)\big) + \rho(\theta_t)\Big)\Big]\mathbb{E}\big[\delta^{\theta_t}\psi_t|\mathcal{F}(t)\big] \\
&\qquad - \zeta_2(t)\Big[1 + 2\lambda\Big(\big(\widehat{\rho}_{t+1} - \rho(\theta_t)\big) + \rho(\theta_t)\Big)\Big]\Big(\delta_t\psi_t - \mathbb{E}\big[\delta_t\psi_t|\mathcal{F}(t)\big]\Big) \\
&\qquad - \zeta_2(t)\Big[1 + 2\lambda\Big(\big(\widehat{\rho}_{t+1} - \rho(\theta_t)\big) + \rho(\theta_t)\Big)\Big]\mathbb{E}\big[(\delta_t - \delta^{\theta_t})\psi_t|\mathcal{F}(t)\big] \\
&\qquad + \zeta_2(t)\lambda\mathbb{E}\big[\epsilon^{\theta_t}\psi_t|\mathcal{F}(t)\big] + \zeta_2(t)\lambda\Big(\epsilon_t\psi_t - \mathbb{E}\big[\epsilon_t\psi_t|\mathcal{F}(t)\big]\Big) + \zeta_2(t)\lambda\mathbb{E}\big[(\epsilon_t - \epsilon^{\theta_t})\psi_t|\mathcal{F}(t)\big]\Bigg).
\end{aligned}$$

By setting $\xi_t = \widehat{\rho}_{t+1} - \rho(\theta_t)$, we may write the above equation as

$$\begin{aligned}
\theta_{t+1} = \Gamma\Bigg(&\theta_t - \zeta_2(t)\big[1 + 2\lambda\big(\xi_t + \rho(\theta_t)\big)\big]\mathbb{E}\big[\delta^{\theta_t}\psi_t|\mathcal{F}(t)\big] \\
&- \zeta_2(t)\big[1 + 2\lambda\big(\xi_t + \rho(\theta_t)\big)\big]\underbrace{\Big(\delta_t\psi_t - \mathbb{E}\big[\delta_t\psi_t|\mathcal{F}(t)\big]\Big)}_{*} \\
&- \zeta_2(t)\big[1 + 2\lambda\big(\xi_t + \rho(\theta_t)\big)\big]\underbrace{\mathbb{E}\big[(\delta_t - \delta^{\theta_t})\psi_t|\mathcal{F}(t)\big]}_{+} \\
&+ \zeta_2(t)\lambda\mathbb{E}\big[\epsilon^{\theta_t}\psi_t|\mathcal{F}(t)\big] + \zeta_2(t)\lambda\underbrace{\Big(\epsilon_t\psi_t - \mathbb{E}\big[\epsilon_t\psi_t|\mathcal{F}(t)\big]\Big)}_{*} + \zeta_2(t)\lambda\underbrace{\mathbb{E}\big[(\epsilon_t - \epsilon^{\theta_t})\psi_t|\mathcal{F}(t)\big]}_{+}\Bigg).
\end{aligned} \tag{18}$$

Since Algorithm 1 uses an unbiased estimator for ρ, we have $\widehat{\rho}_{t+1} \to \rho(\theta_t)$, and thus, $\xi_t \to 0$. The terms $(+)$ asymptotically vanish in lieu of Lemma 8 (Critic convergence). Finally the terms $(*)$ can be seen to vanish using standard martingale arguments (cf. Theorem 2 in [2]). Thus, (18) can be seen to be equivalent in an asymptotic sense to

$$\theta_{t+1} = \Gamma\Big(\theta_t - \zeta_2(t)\big[1 + 2\lambda\rho(\theta_t)\big]\mathbb{E}\big[\delta^{\theta_t}\psi_t|\mathcal{F}(t)\big] + \zeta_2(t)\lambda\mathbb{E}\big[\epsilon^{\theta_t}\psi_t|\mathcal{F}(t)\big]\Big). \tag{19}$$

From Lemma 7 and the foregoing, (17) asymptotically tracks the stable fixed points of the ODE

$$\dot{\theta}_t = \check{\Gamma}\Big(\nabla L(\theta_t, \lambda) + \mathcal{B}(\theta_t, \lambda)\Big). \tag{20}$$

So, if the bias $\sup_\theta \|\mathcal{B}(\theta, \lambda)\| \to 0$, the trajectories (20) converge to those of (6) uniformly on compacts for the same initial condition and the claim follows. ■

Step 3: λ Convergence and Overall Convergence of the Algorithm

The goal here is to first show that the λ-recursion converges and then to prove that the whole algorithm converges to a local saddle point of $L(\theta, \lambda)$. We define the following ODE governing the evolution of λ

$$\dot{\lambda}_t = \check{\Gamma}_\lambda\big(\Lambda(\theta_t) - \alpha\big). \tag{21}$$

Theorem 10 *$\lambda_t \to \mathcal{F}$ almost surely as $t \to \infty$, where $\mathcal{F} \stackrel{\triangle}{=} \{\lambda \mid \lambda \in [0, \lambda_{\max}], \check{\Gamma}_\lambda\big(\Lambda(\theta^\lambda) - \alpha\big) = 0,\ \theta^\lambda \in \mathcal{Z}_\lambda\}$.*

The last step is to establish that the algorithm converges to a (local) saddle point of $L(\theta, \lambda)$, in other words, to a pair (θ^*, λ^*) that are a local minimum w.r.t. θ and a local maximum w.r.t. λ of $L(\cdot, \cdot)$. From Theorem 10, $\lambda_t \to \lambda^*$ for some $\lambda^* \in [0, \lambda_{\max}]$ such that $\theta^{\lambda^*} \in Z_{\lambda^*}$ and $\check{\Gamma}_\lambda\big(\Lambda(\theta^{\lambda^*}) - \alpha\big) = 0$. We now invoke the envelope theorem of mathematical economics [8] to conclude that the ODE (21) is equivalent to $\dot{\lambda}_t = \check{\Gamma}_\lambda\big(\nabla_\lambda L(\theta^{\lambda^*}, \lambda^*)\big)$. From the above, it is clear that (θ_t, λ_t) governed by Eqs. 17 and 18 in the paper converges to a local saddle point of $L(\theta, \lambda)$.

D Extension of the Algorithms to Sharpe Ratio Optimization

D.1 Discounted Setting

The gradient of Sharpe ratio (SR), $S(\theta)$, in the discounted setting is given by

$$\nabla S(\theta) = \frac{1}{\sqrt{\Lambda^\theta(x^0)}}\big(\nabla V^\theta(x^0) - \frac{V^\theta(x^0)}{2\Lambda^\theta(x^0)}\nabla\Lambda^\theta(x^0)\big).$$

The actor recursions for SPSA and SF algorithm variants that optimize the SR objective are as follows:

SPSA

$$\theta_{t+1}^{(i)} = \Gamma_i\Bigg(\theta_t^{(i)} + \frac{\zeta_2(t)}{\sqrt{u_t^\top\phi_u(x^0) - \big(v_t^\top\phi_v(x^0)\big)^2\beta\Delta_t^{(i)}}} \Bigg(\big(v_t^+ - v_t\big)^\top\phi_v(x^0) - \frac{v_t^\top\phi_v(x^0)\big((u_t^+ - u_t)^\top\phi_u(x^0) - 2v_t^\top\phi_v(x^0)(v_t^+ - v_t)^\top\phi_v(x^0)\big)}{2\Big(u_t^\top\phi_u(x^0) - \big(v_t^\top\phi_v(x^0)\big)^2\Big)}\Bigg)\Bigg). \tag{22}$$

SF

$$\theta_{t+1}^{(i)} = \Gamma_i\Bigg(\theta_t^{(i)} + \frac{\zeta_2(t)\Delta_t^{(i)}}{\beta\sqrt{u_t^\top\phi_u(x^0) - \big(v_t^\top\phi_v(x^0)\big)^2}} \Bigg(\big(v_t^+ - v_t\big)^\top\phi_v(x^0) - \frac{v_t^\top\phi_v(x^0)\big((u_t^+ - u_t)^\top\phi_u(x^0) - 2v_t^\top\phi_v(x^0)(v_t^+ - v_t)^\top\phi_v(x^0)\big)}{2\Big(u_t^\top\phi_u(x^0) - \big(v_t^\top\phi_v(x^0)\big)^2\Big)}\Bigg)\Bigg). \tag{23}$$

Note that only the actor recursion changes for SR optimization, while the rest of the updates that include the critic recursions for nominal and perturbed parameters remain the same as before in the SPSA/SF based algorithms. Further, SR optimization does not involve the Lagrange parameter λ, and thus, the proposed actor-critic algorithms are two time-scale (instead of three time-scale like the algorithms in the paper) stochastic approximation algorithms in this case.

Remark 2 *For the SR objective, the proposed SPSA and SF algorithms can be modified to work with only one simulated trajectory of the system. This is because in the SR case, we do not require to tune λ, and thus, the simulated trajectory corresponding to the nominal policy parameter θ is not necessary. In this implementation, the gradient is estimated as $\partial_{\theta^{(i)}} S(\theta) \approx S(\theta + \beta\Delta)/\beta\Delta^{(i)}$ for SPSA and as $\partial_{\theta^{(i)}} S(\theta) \approx (\Delta^{(i)}/\beta)S(\theta + \beta\Delta)$ for SF.*

D.2 Average Setting

The gradient of SR in the average setting is given by

$$\nabla S(\theta) = \frac{1}{\sqrt{\Lambda(\theta)}}\big(\nabla\rho(\theta) - \frac{\rho(\theta)}{2\Lambda(\theta)}\nabla\Lambda(\theta)\big).$$

The actor recursion for the SR variant of the risk-sensitive actor-critic algorithm is as follows:

$$\theta_{t+1} = \Gamma\Big(\theta_t + \frac{\zeta_2(t)}{\sqrt{\widehat{\eta}_{t+1} - \widehat{\rho}_{t+1}^2}}\big(\delta_t\psi_t - \frac{\widehat{\rho}_{t+1}(\epsilon_t\psi_t - 2\widehat{\rho}_{t+1}\delta_t\psi_t)}{2(\widehat{\eta}_{t+1} - \widehat{\rho}_{t+1}^2)}\big)\Big). \tag{24}$$

Note that the rest of the updates, including the average reward, TD errors, and critic recursions are as in the risk-sensitive actor-critic algorithm presented in the paper (see Algorithm 1 in the paper).

E Remarks

Here we list several remarks concerning our proposed discounted and average reward risk-sensitive actor-critic algorithms.

Remark 3 *In the proposed algorithms, the critic uses a TD method to evaluate the policies. These algorithms can be implemented with a Monte-Carlo critic that at each time t computes a sample average of the total discounted rewards corresponding to the nominal θ_t and perturbed $\theta_t + \beta\Delta$ policy parameter. This implementation would be similar to that in [13], except here we use simultaneous perturbation methods to estimate the gradient.*

Remark 4 *Average reward analogues of our simultaneous perturbation algorithms can be developed. These algorithms would estimate the average reward ρ and the square reward η on the faster timescale and use these to estimate the gradient of the performance objective. However, a drawback with this approach, compared to the algorithm proposed in Section 6 of the paper, is the necessity for having two simulated trajectories (instead of one) for each policy update.*

Remark 5 *In the discounted setting, another popular variability measure is the* discounted normalized variance *[5]*

$$\Lambda(\mu) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \big(R_t - \rho_\gamma(\mu)\big)^2\right], \tag{25}$$

where $\rho_\gamma(\mu) = \sum_{x,a} d_\gamma^\mu(x|x^0)\mu(x|a)r(x,a)$ and $d_\gamma^\mu(x|x^0)$ is the discounted visiting distribution of state x under policy μ, defined as

$$d_\gamma^\mu(x|x^0) = (1-\gamma)\sum_{t=0}^{\infty} \gamma^t \Pr(x_t = x|x_0 = x^0;\mu).$$

The variability measure (25) *has close resemblance to the average reward variability measure of Eq. 11 in the paper, and thus, any (discounted) risk measure based on* (25) *can be optimized similar to the corresponding average reward risk measure based on Eq. 11 of the paper. Therefore, we only considered the variability measure of Eq. 1 in the paper for discounted MDPs.*

F Simulation Experiments

We implement the following algorithms using the Green Light District (GLD) simulator [15]:
Average Setting:

- **AC**: This is an actor critic algorithm that minimize the long run average sum of the single-stage cost function $h(x_t)$, without considering any risk criteria. This is similar to Algorithm 1 in Bhatnagar et al. [2].
- **RS-AC**: This is the risk-sensitive actor critic algorithm that attempts to solve (12) and is described in Section 6.

Discounted Setting:

- **SPSA**: This is an actor critic algorithm that minimize the long run discounted sum of the single-stage cost function $h(x_t)$, without considering any risk criteria. This is similar to Algorithm 1 in [1].
- **RS-SPSA**: This is the risk-sensitive actor critic algorithm that attempts to solve (12) and updates according to (8).
- **SF**: This is similar to SPSA algorithm above, except that the gradient estimation scheme used here is based on the smoothed functional technique. The update of the policy parameter in this algorithm is given by

$$\theta_{t+1}^{(i)} = \Gamma_i \left(\theta_t^{(i)} + \zeta_2(t) \Big(\frac{\Delta_t^{(i)}}{\beta} (v_t^+ - v_t)^\top \phi_v(x^0) \Big) \right).$$

- **RS-SF**: This is the risk-sensitive variant of the SF algorithm above and updates the actor according to (9).

The underlying policy that guides the selection of the sign configuration in each of the algorithms above is a parameterized Boltzmann family and has the form

$$\mu_\theta(x,a) = \frac{e^{\theta^\top \phi_{x,a}}}{\sum_{a' \in \mathcal{A}(x)} e^{\theta^\top \phi_{x,a'}}}, \quad \forall x \in \mathcal{X} \ , \ \forall a \in \mathcal{A}. \tag{26}$$

The experiments for each algorithm comprised of the following two phases:

Tuning phase Here each iteration involved the simulation run with the nominal policy parameter θ as well as perturbed policy parameter $\theta + \beta\Delta$. We run the algorithm for 500 iterations, where the run length for a particular policy parameter is 150 steps.

Converged run Following the tuning phase, we obtain the converged policy parameter, say θ^*. In the converged run phase, we perform the simulation with the policy parameter θ^* for 50 iterations, where each iteration involves a simulation run length of 5000 steps. The results reported are averages over these 50 iterations.

The road network used for conducting the experiments is shown in Figure 1. Traffic is added to the network at each time step from the edge nodes, i.e., the nodes labelled **E** in Figure 1. The spawn frequencies specify the rate at which traffic is generated at each edge node and follow the Poisson distribution. The spawn frequencies are set such that the proportion of number of vehicles on the main roads (the horizontal ones in Fig. 1) to those on the side roads is in the ratio $100 : 5$. This setting is close to what is observed in practice and has also been used for instance in Prashanth and Bhatnagar [10, 11]. In all our experiments, we set the weights in the single stage cost function (19) as follows: $r_1 = r_2 = 0.5$ and $r_2 = 0.6, s_2 = 0.4$. For the SPSA and SF based algorithms in the discounted setting, we set the parameter $\delta = 0.1$ and the discount factor $\gamma = 0.9$. The parameter α in the formulations (12) and (2) was set to 20. The step-size sequences are chosen as follows:

$$\zeta_1(t) = \frac{1}{t}, \quad \zeta_2(t) = \frac{1}{t^{0.75}}, \quad \zeta_3(t) = \frac{1}{t^{0.66}}, \qquad t \geq 1. \tag{27}$$

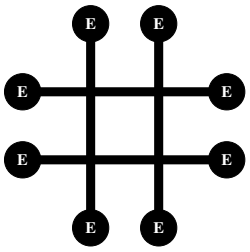


Figure 1: Road Network used for our Experiments.

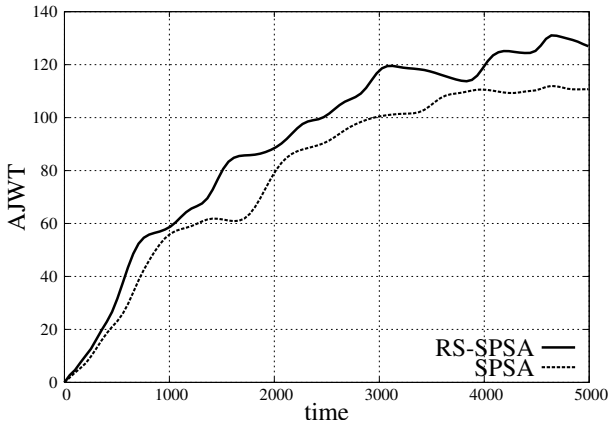


Figure 2: Performance Comparison of SPSA vs. RS-SPSA using the average junction waiting time

Further, the constant k related to $\zeta_4(t)$ is set to 1. It is easy to see that the choice of step-sizes above satisfies (A4). The projection operator γ_i was set to project the iterate θ_i onto the set $[0, 10]$, for all $i = 1, \ldots, \kappa_1$, while the projection operator for the Lagrange multiplier used the set $[0, 1000]$.

We notice from the average junction waiting time plots in Figures. 2 and 3 for the SPSA and SF algorithms and Figure 3(b) (in the main paper) for the average cost algorithm, that the performance of the risk sensitive variants RS-AC, RS-SPSA and RS-SF is close to that of the algorithms AC, SPSA and SF, respectively. As future work, it would be interesting to apply our risk sensitive algorithms in a financial domain and also study the performance of the Sharpe ratio variants discussed in Section D.

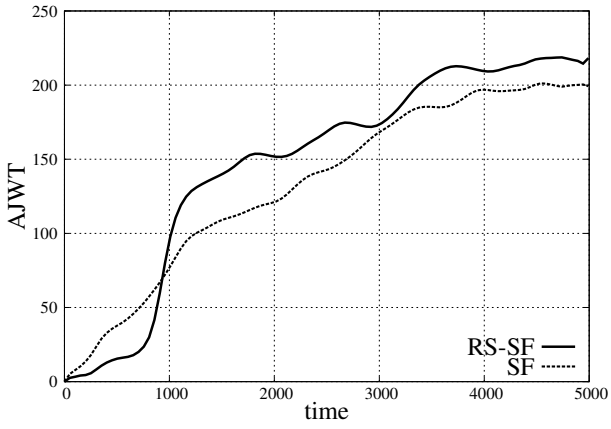


Figure 3: Performance Comparison of SF vs. RS-SF using the average junction waiting time

References

[1] S. Bhatnagar. An actor–critic algorithm with function approximation for discounted cost constrained Markov decision processes. *Systems & Control Letters*, 59(12):760–766, 2010.

[2] S. Bhatnagar, R. Sutton, M. Ghavamzadeh, and M. Lee. Natural actor-critic algorithms. *Automatica*, 45 (11):2471–2482, 2009.

[3] S. Bhatnagar, H. Prasad, and L. Prashanth. *Stochastic Recursive Algorithms for Optimization*, volume 434. Springer, 2013.

[4] V. Borkar. *Stochastic approximation: a dynamical systems viewpoint*. Cambridge University Press, 2008.

[5] J. Filar, L. Kallenberg, and H. Lee. Variance-penalized Markov decision processes. *Mathematics of Operations Research*, 14(1):147–161, 1989.

[6] V. Konda and J. Tsitsiklis. Actor-Critic algorithms. In *Proceedings of Advances in Neural Information Processing Systems 12*, pages 1008–1014, 2000.

[7] H. Kushner and D. Clark. *Stochastic approximation methods for constrained and unconstrained systems*. Springer-Verlag, 1978.

[8] A. Mas-Colell, M. Whinston, and J. Green. *Microeconomic theory*. Oxford University Press, 1995.

[9] J. Peters, S. Vijayakumar, and S. Schaal. Natural actor-critic. In *Proceedings of the Sixteenth European Conference on Machine Learning*, pages 280–291, 2005.

[10] L.A. Prashanth and S. Bhatnagar. Reinforcement Learning With Function Approximation for Traffic Signal Control. *IEEE Transactions on Intelligent Transportation Systems*, 12(2):412 –421, june 2011.

[11] L.A. Prashanth and S. Bhatnagar. Threshold Tuning Using Stochastic Optimization for Graded Signal Control. *IEEE Transactions on Vehicular Technology*, 61(9):3865 –3880, nov. 2012.

[12] R. Sutton, D. McAllester, S. Singh, and Y. Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Proceedings of Advances in Neural Information Processing Systems 12*, pages 1057–1063, 2000.

[13] A. Tamar, D. Di Castro, and S. Mannor. Policy gradients with variance related risk criteria. In *Proceedings of the 29th International Conference on Machine Learning*, pages 387–396, 2012.

[14] A. Tamar, D. Di Castro, and S. Mannor. Temporal difference methods for the variance of the reward to go. In *Proceedings of the 30th International Conference on Machine Learning*, 2013.

[15] M. Wiering, J. Vreeken, J. van Veenen, and A. Koopman. Simulation and optimization of traffic in a city. In *IEEE Intelligent Vehicles Symposium*, pages 453–458, June 2004.