Target-sensitive control of Markov and semi-Markov processes

Abhijit Gosavi

Abstract: We develop the theory for Markov and semi-Markov control using dynamic programming and reinforcement learning in which a form of semi-variance which computes the variability of rewards below a pre-specified target is penalized. The objective is to optimize a function of the rewards and risk where risk is penalized. Penalizing variance, which is popular in the literature, has some drawbacks that can be avoided with semi-variance.

Keywords: target-sensitive, semi-Markov control, semi-variance, relative value iteration, stochastic shortest path problem.

1. INTRODUCTION

Measures used to model risk in the Markov decision process (MDP) include variance [41, 24, 28], exponential utility functions [31, 21, 22, 26, 19, 30, 8, 18, 6], downside risk constraints [13, 44, 45, 25, 2], value at risk [7] and HARA utility functions [35]. Markowitz pioneered the popular use of variance in portfolio models: maximize $\mathsf{E}(\textit{Revenues}) - \theta \mathsf{Var}(\textit{Revenues})$ to keep the risk in check.

There are at least four drawbacks, however, to using variance: (i) variability *above* the mean revenues, which may be desirable, usually gets penalized, (ii) regular variance does not take into account any manager-set targets for meeting revenues, (iii) variance works well usually when the returns have a symmetric distribution, e.g., normal [23], and (iv) in the context of MDPs [24], the resulting variance-penalizing problem has a quadratic structure, solvable via quadratic programming (QP), which makes it difficult to use dynamic programming (DP). We use a different measure of risk called target semi-variance risk (see [36] for an early definition), which measures variability *only below* a target revenue. It can be handily incorporated within the following Markowitz framework, which is commercially popular:

$$\mathsf{E}(\textit{Revenues}) - \theta \mathsf{SemiVar}(\textit{Revenues})$$

More importantly, as we will show, unlike variance for which one must use QP [24], one *can* develop a DP and also an reinforcement learning (RL) framework for solving the resultant problem, since some of its properties follow directly from those of the average reward problem.

The target semi-variance or the "semi-variance" metric is of independent interest in economics because of applications in finance. In particular, it leads to a new type of behavior called mean-semi-variance equilibria [23], which has some attractive properties of the expected utility of risk. Also, semi-variance is related to downside risk, i.e., the probability of the revenues falling below a specified target and is popular in optimal hedging (see [43]).

The contributions of this paper are as follows. We present a new value iteration algorithm for the semi-MDP (SMDP) and show the convergence of the relative value iteration algorithm for the MDP. Although the case of semi-variance penalties can be studied as a special case of the classical average reward problem, we will show that the SMDP value-iteration algorithm that we develop here does not require discretization needed for the value iteration approach in the average reward case [5]. For the MDP, value iteration is known to converge in the span; we show in this paper that *relative* value iteration, which is numerically more stable than value iteration, also converges in the span. For problems in which transition probabilities are not available due to a complex large-scale system, we present some *new* analysis for the RL algorithms for MDP and the SMDP. In this paper, our main contribution is to analyze the use of dynamic programming and its variants for semi-variance penalties. In [29], we study its use on an industrial problem using a linear programming approach.

The rest of this paper is organized as follows. Some definitions are presented in Section 2. DP algorithms are discussed in Section 3. RL algorithms are developed in Section 4. Our computational results are in Section 5. Concluding remarks are in Section 6.

2. SEMI-VARIANCE

We will first define the semi-Markov case. Let $\mathscr{A}(i)$ denote the finite set of actions allowed in state i, $\mathscr{S}$ the finite set of states, and $\mu(i)$ the action chosen in state i when policy μ is followed, where $\cup_{i\in\mathscr{S}}\mathscr{A}(i) = \mathscr{A}$. Also, let $r(.,.,.) : \mathscr{S} \times \mathscr{A} \times \mathscr{S} \rightarrow \Re$ and $t(.,.,.) :$

A. Gosavi is with the Department of Engineering Management, Missouri University of Science and Technology, 219 Engineering Management, Rolla, MO 65401, USA; E-mail: gosavia@mst.edu

$\mathscr{S} \times \mathscr{A} \times \mathscr{S} \to \Re$ denote the reward in one transition and the time in one transition, respectively, and $p(.,.,.) : \mathscr{S} \times \mathscr{A} \times \mathscr{S} \to [0,1]$ denote the associated transition probability. Then the *expected* immediate reward earned in state i when action a is selected in it is: $\bar{r}(i,a) = \sum_{j=1}^{|\mathscr{S}|} p(i,a,j) r(i,a,j)$ and the expected immediate time of transition from state i under action a is:

$$\bar{t}(i,a) = \sum_{j=1}^{|\mathscr{S}|} p(i,a,j) t(i,a,j).$$

We will assume that every Markov chain in the problem is regular. For the SMDP, the long-run average reward of a policy μ starting at state i is: $\rho_\mu(i) \equiv \frac{\liminf_{k\to\infty} E_\mu\left[\sum_{s=1}^{k} \bar{r}(x_s,\mu(x_s))|x_1=i\right]/k}{\limsup_{k\to\infty} E_\mu\left[\sum_{s=1}^{k} \bar{t}(x_s,\mu(x_s))|x_1=i\right]/k}$, where x_s is the state occupied before the sth transition and E_μ is the expectation induced by μ. For the Markov case, the definition can be obtained by replacing the function $t(.,.,.)$ by 1. If the Markov chain of every policy is regular, it can be shown that the average reward is independent of the starting state.

In our notation, $\vec{x}$ will denote a column vector whose ith element is $x(i)$. P_μ will denote the transition probability matrix of μ. Let $v(.,.,.) : \mathscr{S} \times \mathscr{A} \times \mathscr{S} \to \Re$ denote the semi-variance, which would be defined as follows for $i \in \mathscr{S}$, $j \in \mathscr{S}$, and $a \in \mathscr{A}(i)$: $v(i,a,j) = \left[\{\tau t(i,a,j) - r(i,a,j)\}_+\right]^2$, where τ denotes the pre-set target value for average reward per unit time (or simply the "target") and $\{a\}_+ = \max(0,a)$. Also, for any $i \in \mathscr{S}$ and $a \in \mathscr{A}(i)$, we define $\bar{v}(i,a) = \sum_{j=1}^{|\mathscr{S}|} p(i,a,j) v(i,a,j)$ and for a given positive scalar θ, $w(i,a,j) = r(i,a,j) - \theta v(i,a,j)$ and $\bar{w}(i,a) = \bar{r}(i,a) - \theta \bar{v}(i,a)$.

Definition 1. The long-run semi-variance of μ starting at i for the SMDP

$$\kappa_\mu^2(i) \equiv \frac{\liminf_{k\to\infty} E_\mu\left[\sum_{s=1}^{k} \bar{v}(x_s,\mu(x_s))|x_1=i\right]/k}{\limsup_{k\to\infty} E_\mu\left[\sum_{s=1}^{k} \bar{t}(x_s,\mu(x_s))|x_1=i\right]/k}.$$

Let $\vec{v_\mu}$ and $\vec{t_\mu}$ denote the column vectors whose ith element is $\bar{v}(i,\mu(i))$ and $\bar{t}(i,\mu(i))$ respectively. Then, from the definition above, it follows that: $\vec{\kappa_\mu^2} = \frac{\left(\lim_{k\to\infty} \frac{1}{k} \sum_{m=0}^{k-1} P_\mu^k\right)\vec{v_\mu}}{\left(\lim_{k\to\infty} \frac{1}{k} \sum_{m=0}^{k-1} P_\mu^k\right)\vec{t_\mu}}$. Since $\lim_{k\to\infty} \frac{1}{k} \sum_{m=0}^{k-1} P_\mu^k$ exists for regular Markov chains, it follows that: $\kappa_\mu^2(j) = \frac{\sum_{i\in\mathscr{S}} \Pi_\mu(i) \bar{v}(i,\mu(i))}{\sum_{i\in\mathscr{S}} \Pi_\mu(i) \bar{t}(i,\mu(i))}$ for any $j \in \mathscr{S}$, and that κ^2 for a given policy does not depend on the starting state. Then our objective function, or semi-variance-penalized score, for a policy μ in the SMDP, with all policies having regular chains, is: $\phi_\mu \equiv$

$$\begin{aligned} \rho_\mu - \theta \kappa_\mu^2 &= \frac{\sum_{i\in\mathscr{S}} \Pi_\mu(i) \left[\bar{r}(i,\mu(i) - \theta \bar{v}(i,\mu(i))\right]}{\sum_{i\in\mathscr{S}} \Pi_\mu(i) \bar{t}(i,\mu(i))} \quad (1) \\ &= \frac{\sum_{i\in\mathscr{S}} \Pi_\mu(i) \bar{w}(i,\mu(i))}{\sum_{i\in\mathscr{S}} \Pi_\mu(i) \bar{t}(i,\mu(i))}. \end{aligned}$$

Definition 2. If $\vec{h}$ denotes a vector in $\Re^{|\mathscr{S}|}$, then we define the transformation L_μ as: $\forall i \in \mathscr{S}, L_\mu h(i) =$

$$\sum_{j\in\mathscr{S}} p(i,\mu(i),j) \left[r(i,\mu(i),j) - \theta v(i,\mu(i),j) + h(j)\right]$$

and $\forall i \in \mathscr{S}, Lh(i) =$

$$\max_{a\in\mathscr{A}(i)} \left[\sum_{j\in\mathscr{S}} p(i,a,j) \left[r(i,a,j) - \theta v(i,a,j) + h(j)\right]\right]. \quad (2)$$

The following result can be obtained from the classical result for average reward (Prop. 4.2.1 in [5]) via replacing the reward function $r(.,.,.)$ by $w(.,.,.)$.

Theorem 1: (Poisson equation) If $\phi \in \Re$ and $\vec{h} \in \Re^{|\mathscr{S}|}$ satisfy for all $i \in \mathscr{S}$:

$$h(i) = \sum_{j\in\mathscr{S}} p(i,\mu(i),j)[r(i,\mu(i),j) - \theta v(i,\mu(i),j) - \phi t(i,a,j) + h(j)], \quad (3)$$

then ϕ is the semi-variance score associated with the policy μ. Furthermore (Bellman equation) if a scalar $\phi^* \in \Re$ and $\vec{J} \in \Re^{|\mathscr{S}|}$ satisfy for all $i \in \mathscr{S}$

$$J(i) = \max_{u\in\mathscr{A}(i)} [\sum_{j\in\mathscr{S}} p(i,u,j)[r(i,u,j) - \theta v(i,u,j) - \phi^* t(i,a,j) + J(j)]], \quad (4)$$

then ϕ^* is the semi-variance score associated with the policy μ^* that attains the max in the RHS of Equation (4). The policy μ^* is the optimal policy.

3. DP ALGORITHMS

MDP: We first present a value iteration (VI) and a relative VI (RVI) algorithm for the MDP. Note that the span of vector $\vec{x}$: $sp(\vec{x}) = \max_i x(i) - \min_i x(i)$.

Steps in VI: Step 1: Set $k = 0$ and select an arbitrary vector $\vec{J}^0$. Specify $\varepsilon > 0$.

Step 2: For each $i \in \mathscr{S}$, compute:

$$J^{k+1}(i) = \max_{a\in\mathscr{A}(i)} \left[\bar{r}(i,a) - \theta \bar{v}(i,a) + \sum_j p(i,a,j) J^k(j)\right].$$

Step 3: If $sp(\vec{J}^{k+1} - \vec{J}^k) < \varepsilon$, go to Step 4. Otherwise increase k by 1 and return to Step 2.

Step 4: The policy returned by the algorithm is d, which is computed as follows: For each $i \in \mathscr{S}$ choose,

$$d(i) \in \arg\max_{a\in\mathscr{A}(i)} \left[\bar{r}(i,a) - \theta \bar{v}(i,a) + \sum_{j\in\mathscr{S}} p(i,a,j) J^k(j)\right]. \quad (5)$$

Steps in RVI: Step 1: Set $k = 0$, choose any state to be a distinguished state, i^*, and select an arbitrary vector $\vec{R}^0$.

Specify $\varepsilon > 0$ and set $\vec{R}^{\,0} = \vec{J}^{\,0} - J^0(i^*)\vec{e}$, where $\vec{e}$ is a column vector of ones.
Step 2: For each $i \in \mathscr{S}$, compute: $J^{k+1}(i) =$

$$\max_{a\in\mathscr{A}(i)} \left[\bar{r}(i,a) - \theta \bar{v}(i,a) + \sum_{j\in\mathscr{S}} p(i,a,j) R^k(j) \right].$$

Then for each $i \in \mathscr{S}$, set $R^{k+1}(i) = J^{k+1}(i) - J^{k+1}(i^*)$.
Step 3: If $sp(\vec{R}^{\,k+1} - \vec{R}^{\,k}) < \varepsilon$, go to Step 4; else increase k by 1 and return to Step 2.
Step 4: For each $i \in \mathscr{S}$ choose

$$d(i) \in \arg\max_{a\in\mathscr{A}(i)} \left[\bar{r}(i,a) - \theta \bar{v}(i,a) + \sum_{j\in\mathscr{S}} p(i,a,j) R^k(j) \right]$$

and stop. The policy returned by the algorithm is d.

For analyzing the RVI algorithm, we need the following basic result (see e.g., Theorems 6.6.2 and 8.5.2 [37]).

Theorem 2: Suppose F is an M-stage span contraction mapping; that is, for any two vectors $\vec{x}$ and $\vec{y}$ in a given vector space, for some positive, finite, and integral value of M,

$$sp(F^M\vec{x} - F^M\vec{y}) \le \eta \; sp(\vec{x} - \vec{y}) \text{ for } 0 \le \eta < 1. \quad (6)$$

Consider the sequence $\{\vec{z}^{\,k}\}_{k=1}^{\infty}$ defined by:
$\vec{z}^{\,k+1} = F\vec{z}^{\,k} = F^{k+1}\vec{z}^{\,0}$. Then, there exists a $\vec{z}^{\,*}$

$$\text{for which } sp(F\vec{z}^{\,*} - \vec{z}^{\,*}) = 0 \text{ and} \quad (7)$$

$$\lim_{k\to\infty} sp(\vec{z}^{\,k} - \vec{z}^{\,*}) = 0. \quad (8)$$

Also, given an $\varepsilon > 0$, there exists an N such that for all $k \ge N$:

$$sp(\vec{z}^{\,kM+1} - \vec{z}^{\,kM}) < \varepsilon. \quad (9)$$

We denote the delta coefficient of a matrix $\mathbf{A}$ by $\alpha_{\mathbf{A}}$ (see Appendix for definition). We prove convergence of the RVI algorithm in span under a condition for the matrix in (10). The convergence of the VI algorithm in span has been shown under a different condition in Theorems 8.5.2 and 8.5.3 of [37]; the following result (Theorem 3) also holds under the condition of [37]. See the Appendix for Lemma 2 and Lemma 3 (inspired by a result in [32]) which are needed below.

Theorem 3: (a). Consider a pair of finite sequences of M stationary policies, $S_1 = \{\mu_1, \mu_2, \ldots \mu_M\}$ and $S_2 = \{\nu_1, \nu_2, \ldots \nu_M\}$. Further, consider the stacked matrix:

$$\mathbf{A}_{S_1,S_2} \equiv \left[\frac{\mathbf{P}_{\mu_1} \cdot \mathbf{P}_{\mu_2} \cdots \mathbf{P}_{\mu_M}}{\mathbf{P}_{\nu_1} \cdot \mathbf{P}_{\nu_2} \cdots \mathbf{P}_{\nu_M}} \right]. \quad (10)$$

Assume that there exists an integral value for $M \ge 1$ such that for every possible pair, (S_1, S_2), the delta coefficient of $\mathbf{A}_{S_1,S_2}$ is less than 1. Then, the VI algorithm converges in the limit to an optimal solution.
(b). The VI and RVI algorithms choose the same sequence of maximizing actions and terminate at the same policy for a given value of ε.

Proof: (a). Consider the sequence of vectors $\vec{J}^{\,k}$ in VI. Then: $\vec{J}^{\,k+1} = L\vec{J}^{\,k}$, for all $k = 1, 2, \ldots$ where L, defined in (2), is the transformation used in Step 2 of VI. The delta-coefficient condition in the assumption above implies that Lemma 3, proved in Appendix, is true, from which one has that Theorem 2 holds for L. It follows from (7) then that there exists a $\vec{J}^*$ such that $L\vec{J}^* = \vec{J}^* + \psi_1\vec{e}$ for some scalar ψ_1. The above implies from Theorem 1 (setting $t(.,.,.) = 1$) that $\vec{J}^*$ is an optimal solution of the MDP. However, from (8), we know that $\lim_{k\to\infty} \vec{J}^{\,k} = \vec{J}^* + \psi_2\vec{e}$ for some scalar ψ_2. From (5), one has that $\vec{J}^*$ and $(\vec{J}^* + \psi_2\vec{e})$ will result in the same policy. It follows from (9) that a finite termination rule can be developed with a user-specified value of ε.
(b). Let $\vec{R}^{\,k}$ denote the iterate vector in the kth iteration of RVI. We will first show that:

$$\vec{R}^{\,k} = \vec{J}^{\,k} - \sum_{l=1}^{k} x^l\vec{e}, \quad (11)$$

where x^l is a scalar constant whose value depends on iteration l and $\vec{e}$ is a vector of ones. We will use induction on k. It is clear from Step 1 of RVI that: $\vec{R}^{\,1} = \vec{J}^{\,1} - x^1\vec{e}$, where $x^1 = J^1(i^*)$, and hence (11) is true for $k = 1$. Assuming it is true for $k = m$, we have that:

$$\vec{R}^{\,m} = \vec{J}^{\,m} - \sum_{l=1}^{m} x^l\vec{e}. \quad (12)$$

Now, since $w(i,a,j) = r(i,a,j) - \theta v(i,a,j)$, from Step 2 of RVI, by setting $x^{m+1} = J^{m+1}(i^*)$, we have that for all $i \in \mathscr{S}, R^{m+1}(i)$:

$$\begin{aligned}
&= \max_{j\in\mathscr{S}} \left(\sum_{j\in\mathscr{S}} p(i,a,j) \left[w(i,a,j) + R^m(j) \right] \right) - x^{m+1} \\
&= \max_{j\in\mathscr{S}} \left(\sum_{j\in\mathscr{S}} p(i,a,j) \left[w(i,a,j) + J^m(j) - \sum_{l=1}^{m} x^l \right] \right) \\
&\quad - x^{m+1} \text{ (from (12))} \\
&= \max_{j\in\mathscr{S}} \left(\sum_{j\in\mathscr{S}} p(i,a,j) \left[w(i,a,j) + J^m(j) \right] \right) \\
&\quad - \sum_{l=1}^{m+1} x^l \qquad (13) \\
&= J^{m+1}(i) - \sum_{l=1}^{m+1} x^l \text{ (from Step 2 of VI)},
\end{aligned}$$

from which (11) follows for any k. The span of the difference vector in any iteration of both algorithms will be equal, since: $sp\left(\vec{R}^{\,k+1} - \vec{R}^{\,k}\right)$

$$\begin{aligned}
&= sp\left(\vec{J}^{\,k+1} - \vec{J}^{\,k} - x^{m+1}\vec{e}\right) \text{ (from (11))} \\
&= sp\left(\vec{J}^{\,k+1} - \vec{J}^{\,k}\right).
\end{aligned}$$

The two algorithms will choose the same sequence of maximizing actions (see Step 4 in each) since:

$$\begin{aligned} & \arg\max_{j\in\mathscr{S}} \left[\sum_{j\in\mathscr{S}} p(i,a,j)\left[r(i,a,j)+R^k(j)\right]\right] \\ = & \arg\max_{j\in\mathscr{S}} \left[\sum_{j\in\mathscr{S}} p(i,a,j)\left[r(i,a,j)+J^k(j)-\sum_{l=1}^{k} x^l\right]\right] \\ = & \arg\max_{j\in\mathscr{S}} \left[\sum_{j\in\mathscr{S}} p(i,a,j)\left[r(i,a,j)+J^k(j)\right]\right]. \end{aligned}$$

□

A note about notation: In a proof below, we will use the following shorthand notation

$$\vec{x} \geq \vec{y}$$

for column vectors, $\vec{x}$ and $\vec{y}$, to indicate that for *every* row i in the vector:

$$x(i) \geq y(i).$$

Thus, in the two-dimensional vector case this will imply:

$$x(1) \geq y(1) \text{ and } x(2) \geq y(2).$$

The above notation will also be used in the context of $\leq$, $<$, and $>$.

SMDP: The average reward MDP can be solved by solving an associated stochastic shortest path, abbreviated as SSP, problem (see Prop 7.4.1 in Vol I of [5]). We will first extend this key result to the SMDP.

Lemma 1: Consider any recurrent state in the SMDP, and number it, n. Consider an SSP with the same Markov chains. Introduce a fictitious state, s, in the SSP, and set $p(i,a,n) = 0$ and $p(i,a,s) = p(i,a,n)$ for all $a \in \mathscr{A}(i)$. Let $R_n(\mu)$ and $T_n(\mu)$ denote the expected total reward and the expected total time, respectively, in one cycle from n to n if the policy pursued is μ. Then define $\hat{\rho} \equiv \max_\mu \frac{R_n(\mu)}{T_n(\mu)}$. Assume that the immediate reward in the SSP is $r(i,a,j) - \hat{\rho}t(i,a,j)$. Then $\hat{\rho}$ equals the optimal long-run reward per unit time of the SMDP.

Proof: We first define the expected reward and the expected time in one cycle as follows:

$$E[R_n(\mu)] \equiv \bar{R}_\mu; \quad E[T_n(\mu)] \equiv \bar{T}_\mu.$$

Let $n = |\mathscr{S}|$ without loss of generality. Via Prop 7.2.1 b in [5] (Vol I), we have that $h^*(i)$ for $i = 1,2,\ldots,n$, solves the Bellman equation for the SSP:

$$h^*(i) = \max_{a\in\mathscr{A}(i)} \left[\bar{r}(i,a) - \hat{\rho}\bar{t}(i,a) + \sum_{j=1}^{n-1} p(i,a,j)h^*(j)\right].$$

In the RHS of the above, we omit $h^*(n)$ since $p(i,a,n) = 0$. By its definition: $h_\mu(i) =$

$$\lim_{N\to\infty} \frac{1}{N} \mathsf{E}\left[f(x,\mu,N)|x_1 = i\right]$$

where

$$f(x,\mu,N) = \sum_{k=1}^{N} r(x_k,\mu(x_k),x_{k+1}) - \hat{\rho}t(x_k,\mu(x_k),x_{k+1}),$$

and $\mathsf{E}[.]$ is an expectation over a random trajectory; when $x_{k+1} = n$, we set $x_{k+2} = i$. Then, when $i = n$, we have that the summation in the above is over cycles of the SSP, and hence

$$h_\mu(n) = R_n(\mu) - \hat{\rho}T_n(\mu),$$

which implies $h^*(n) =$

$$\max_\mu [R_n(\mu) - \hat{\rho}T_n(\mu)] = \max_\mu \left[\frac{R_n(\mu)}{T_n(\mu)} - \hat{\rho}\right] T_n(\mu) = 0.$$

The above implies that $h^*(n) = 0$ and for $i = 1,2,\ldots,n-1$

$$h^*(i) = \max_{a\in\mathscr{A}(i)} \left[\bar{r}(i,a) - \hat{\rho}\bar{t}(i,a) + \sum_{j=1}^{n} p(i,a,j)h^*(j)\right]. \quad (14)$$

We define $\vec{J}_0 = \vec{h}^*$ and for some stationary policy μ, using $\vec{r}_\mu$ to denote the vector whose ith element is $\bar{r}(i,\mu)$:

$$\vec{J}_{k+1} = \vec{r}_\mu + P_\mu \vec{J}_k. \quad (15)$$

Then, if $\vec{\tau}_\mu$ denotes the vector whose ith element is $\bar{t}(i,\mu(i))$, we will show via induction that:

$$\hat{\rho}\sum_{k=1}^{m} P_\mu^k \vec{\tau}_\mu + \vec{J}_0 \geq \vec{J}_m. \quad (16)$$

Now, from Eqn. (14), for any given stationary policy μ,

$$\vec{h}_* \geq \vec{r}_\mu - \hat{\rho}P_\mu\vec{\tau}_\mu + P_\mu\vec{h}_*, \text{i.e., } \vec{J}_0 \geq \vec{r}_\mu - \hat{\rho}P_\mu\vec{\tau}_\mu + P_\mu\vec{h}_*$$

from which we have, using (15),

$$\hat{\rho}P_\mu\vec{\tau}_\mu + \vec{J}_0 \geq \vec{r}_\mu + P_\mu\vec{h}_* = \vec{r}_\mu + P_\mu\vec{J}_0 = \vec{J}_1; \text{ i.e.,}$$

$$\hat{\rho}P_\mu\vec{\tau}_\mu + \vec{J}_0 \geq \vec{J}_1. \quad (17)$$

Multiplying both sides of the above by P_μ and then adding $\vec{r}_\mu$ to both sides, we obtain:

$$\hat{\rho}P_\mu^2\vec{\tau}_\mu + P_\mu\vec{J}_0 + \vec{r}_\mu \geq P_\mu\vec{J}_1 + \vec{r}_\mu, \text{ which using (15)}$$

$$\text{becomes } \hat{\rho}P_\mu^2\vec{\tau}_\mu + \vec{J}_1 \geq \vec{J}_2. \quad (18)$$

Adding (17) and (18), we have: $\hat{\rho}\sum_{k=1}^{2} P_\mu^k\vec{\tau}_\mu + \vec{J}_o \geq \vec{J}_m$, i.e., (16) holds for $m = 2$. We now assume (16) to be true

for $m = l$, then multiply both of its sides by P_μ and then add $\bar{r}_\mu$ to both sides to obtain:

$$\hat{\rho}\sum_{k=1}^{l} P_\mu^{k+1}\vec{\tau}_\mu + P_\mu\vec{J}_0 + \bar{r}_\mu \geq P_\mu\vec{J}_l + \bar{r}_\mu, \text{ which results in}$$

$$\hat{\rho}\sum_{k=1}^{l} P_\mu^{k+1}\vec{\tau}_\mu + \vec{J}_1 \geq \vec{J}_{l+1}. \tag{19}$$

Adding (19) and (17), we have $\hat{\rho}\sum_{k=1}^{l+1} P_\mu^k\vec{\tau}_\mu + \vec{J}_0 \geq \vec{J}_{l+1}$, which completes the induction. Then dividing both sides of (16) by m and taking the limit as $m \to \infty$, we have:

$$\hat{\rho}\lim_{m\to\infty}\frac{\sum_{k=1}^{m} P_\mu^k\vec{\tau}_\mu}{m} + \lim_{m\to\infty}\frac{\vec{J}_0}{m} \geq \lim_{m\to\infty}\frac{\vec{J}_m}{m}. \tag{20}$$

Now, from Prop. 4.1.1 of vol II of [5], using $\vec{e}$ to denote a column vector whose every element is 1,

$$\lim_{m\to\infty}\frac{\sum_{k=1}^{m} P_\mu^k\vec{\tau}_\mu}{m} = \left(\sum_{i\in\mathscr{S}}\Pi_\mu\tau_\mu(i)\right)\vec{e} \equiv \bar{T}_\mu\vec{e}.$$

Also, from section 4.3 of vol II of [5],

$$\lim_{m\to\infty}\frac{\vec{J}_m}{m} = \bar{R}_\mu\vec{e}$$

Since $J_0(i)$ is finite for every i,

$$\lim_{m\to\infty}\frac{\vec{J}_0}{m} = 0\vec{e}.$$

Then, using the above, we can write (20) as:

$$\hat{\rho}\bar{T}_\mu\vec{e} \geq \bar{R}_\mu\vec{e}, \text{ i.e., } \hat{\rho}\vec{e} \geq \frac{\bar{R}_\mu}{\bar{T}_\mu}\vec{e}. \tag{21}$$

Since, the numerator in the RHS of (21) equals the expected reward in one transition under μ and the denominator denotes the expected time in one transition, via the renewal reward theorem [38], the RHS of (21) is ρ_μ, the long-run reward per unit time of the policy μ, i.e., $\hat{\rho} \geq \rho_\mu$. The equality in (21) applies when one uses the policy μ^*, whose average reward is ρ^*, that uses the max operator in (14). Then μ^* is optimal for the SMDP (via Prop 5.3.1 in [5], Vol II); i.e., $\hat{\rho} = \rho_{\mu^*} = \rho^*$. □

We can apply the above result for our semi-variance-penalized case by replacing the function r by w. We now present an action-value-based DP algorithm for the SMDP. It is derived from the two-time-scale RL algorithm for the MDP in [1]. We note that the algorithm below solves the SMDP *without* discretization (discretizing the SMDP); the existing value iteration approach in the literature for SMDPs (see Prop. 5.3.3 in vol II of [5]) requires discretization. The main update of the algorithm is that for all (i,a) pairs,

$$Q^k(i,a) = (1-\alpha^k)Q^k(i,a) +$$

$$\alpha^k\left[\sum_{j\in\mathscr{S}} p(i,a,j)\left\{r(i,a,j) - \theta v(i,a,j) - \phi^k t(i,a,j)\right\}\right]$$

$$+\alpha^k\left[\sum_{j\neq i^*} p(i,a,j)\max_{b\in\mathscr{A}(j)} Q(j,b)\right],$$

and after one update of all action-values, update $\phi^{k+1} = \Pi\left[\phi^k + \beta^k \max_{b\in\mathscr{A}(i^*)} Q(i^*,b)\right]$ where i^* is a special state that can be any state in the system (we assume that all Markov chains are regular) and $\Pi[.]$ is a projection onto $[-L,L]$ where $L = \max_{i,j,a}|w(i,a,j)|$. The step sizes, α and β, in addition to the usual conditions (sums equal infinity and sums of squares equal a finite number), must satisfy: $\lim_{k\to\infty}\beta^k/\alpha^k = 0$. We define $Q^*(i,a)$ to be the optimal Q-value for (i,a) pair that solves the Q-version of the Bellman equation (4). In the above, one essentially solves the equivalent SSP for the SMDP.

Theorem 4: $Q^k(i,a)$ tends to $Q^*(i,a)$ for all (i,a) and ϕ^k tends to ϕ^* almost surely as k tends to ∞.

Proof: Using Lemma 1 to invoke the connection with the SMDP and noting that the transformation underlying the main update has the weighted norm contraction, the result follows directly from Theorem 4.5 of [1] after setting their martingale noise terms to 0. □

4. REINFORCEMENT LEARNING ALGORITHMS

Vanishing Discount Algorithm: We now present for the MDP a single time scale algorithm that uses the vanishing discount approach. Given a sequence $\{\lambda_k\}_{k=1}^{\infty}$, for all $i \in \mathscr{S}$ consider $V_\lambda(i) =$

$$\max_{a\in\mathscr{A}(i)}\left\{\sum_{j\in\mathscr{S}} p(i,a,j)\left[r(i,a,j) - \theta v(i,a,j) + \lambda V_\lambda(j)\right]\right\}. \tag{22}$$

Now if we fix a state i^* and define for all $i \in \mathscr{S}$, $J_\lambda(i) = V_\lambda(i) - V_\lambda(i^*)$, the above becomes:

$$(1-\lambda)V_\lambda(i^*) + J_\lambda(i) =$$

$$\max_{a\in\mathscr{A}(i)}\left\{\sum_{j\in\mathscr{S}} p(i,a,j)\left[r(i,a,j) - \theta v(i,a,j) + \lambda J_\lambda(j)\right]\right\}. \tag{23}$$

Theorem 6.18 and Corollary 6.20 in Ross [38] together imply that: for all $i \in \mathscr{S}$, there exists $\lim_{\lambda\to 1} J_\lambda(i) = J(i)$ and $\lim_{\lambda\to 1}(1-\lambda)V_\lambda(i^*) = \phi^*$, where ϕ^* is some constant. It then follows that as λ_k tends to 1, Equation (23) becomes Equation (4). Now we define for all (i,a):

$$Q(i,a) = \sum_{j\in\mathscr{S}} p(i,a,j)\left[r(i,a,j) - \theta v(i,a,j) + \lambda V_\lambda(j)\right], \tag{24}$$

where $\vec{V}_\lambda$ denotes the optimal value function for a given value of λ. Equations (22) and (24) imply that for every $i \in \mathscr{S}$, $V_\lambda(i) = \max_{a \in \mathscr{A}(i)} Q(j,a)$, which from (24) implies that $Q(i,a) =$

$$\sum_{j \in \mathscr{S}} p(i,a,j) \left[r(i,a,j) - \theta v(i,a,j) + \lambda \max_{b \in \mathscr{A}(j)} Q(j,b) \right]. \quad (25)$$

This motivates the following RL algorithm (along the standard lines of Q-Learning):

$$Q(i,a) \leftarrow (1-\alpha) Q(i,a) +$$

$$\alpha \left[r(i,a,j) - \theta v(i,a,j) + \lambda \max_{b \in \mathscr{A}(j)} Q(j,b) \right]. \quad (26)$$

Theorem 5: The RL algorithm in (26) converges to an optimal solution of the semi-variance-penalized problem almost surely, as λ tends to 1.

Proof: The transformation in Equation (25) can be shown to be contractive, and hence it has a unique fixed point and it is non-expansive. Then, if the iterate $Q(.,.)$ remains bounded, convergence to the fixed point of the transformation follows from Lemma 2.4 in [10]. We now need to show that the iterate $Q(i,a)$ is bounded.

Setting, $m = (i,a)$, we define $g : \Re^n \rightarrow \Re^n$ as:

$$g_{i,a}(\vec{Q}) = -Q(i,a) +$$

$$\sum_{j \in \mathscr{S}} p(i,a,j) \left[r(i,a,j) - \theta v(i,a,j) + \lambda \max_{b \in \mathscr{A}(j)} Q(j,b) \right].$$

Define a scaled function $g^\zeta : \Re^n \rightarrow \Re^n$ as: $g^\zeta(\vec{x}) = \frac{g(\zeta \vec{x})}{\zeta}$. It follows then that g^∞ exists and is given as:

$$g^\infty_{i,a}(\vec{Q}) = \lambda \sum_{j \in \mathscr{S}} p(i,a,j) \max_{b \in \mathscr{A}(j)} Q(j,b) - Q(i,a).$$

It also follows that g is Lipschitz continuous. We rewrite the matrix $g^\infty(.)$ as: $g^\infty(\vec{Q}) = \lambda \mathbf{P}' - \mathbf{I}$, where $\mathbf{P}'$ is a matrix whose every element is a transition probability or 0, and $\mathbf{I}$ is the identity matrix; the above follows from the definition of $g^\infty(.)$. Since, $\lambda < 1$, we have that $||\lambda \mathbf{P}'||_\infty < 1$, and hence $\sigma(\lambda \mathbf{P}') < 1$, where $\sigma(.)$ denotes the spectral radius. If ψ denotes an eigenvalue of $\lambda \mathbf{P}'$, then the above implies: $|\psi| < 1$. Now, the eigenvalue of $\lambda \mathbf{P}' - \mathbf{I}$ must equal the eigenvalue of $\lambda \mathbf{P}'$ minus 1, and hence the eigenvalue of $\lambda \mathbf{P}' - \mathbf{I}$ must be strictly negative. This implies from a basic result in linear systems (see Theorem 4.1 on page 151 of [15]) that the zero solution of the ODE, $\frac{d\vec{x}(t)}{dt} = g^\infty(\vec{x}(t))$, must be asymptotically stable. This implies from Theorem 2.1 (i) of [11] that the iterate $Q(i,a)$ remains bounded.

The unique fixed point to which the values converge is the optimal solution of the λ-discounted problem, i.e., solution of (25) or (22) in terms of the value function. As λ tends to 1, as argued above, we obtain a policy that satisfies a Q-version of (4). From Theorem 1 (setting $t(.,.,.) = 1$), this is the optimal solution. □

For the SMDP, under discounting, where $w_R(i,a,j)$ denotes the continuous rate of variance-penalized reward from state i to j under action a, and $w_I(i,a,j)$ denotes the variance-penalized reward earned from *i immediately* after action a is taken and the system goes to j, the Bellman equation (vol II of [5]) is:

$$J(i) = \max_{a \in \mathscr{A}(i)} \left[\sum_{j \in S} p(i,a,j) w_I(i,a,j) + W(i,a) + \sum_j \int_0^\infty \exp(-\tilde{\lambda} t) J(j) f_{i,a,j}(t) dt \right],$$

where $\tilde{\lambda}$ is the continuous discount factor,

$$W(i,a) = \sum_j \int_0^\infty w_R(i,a,j) \frac{1 - \exp(-\tilde{\lambda} t)}{\tilde{\lambda}} f_{i,a,j}(t) dt$$

and $\lim_{t \rightarrow \infty} f_{i,a,j}(t) = p(i,a,j)$. This suggests the following RL algorithm: For all (i,a),

$$Q(i,a) \leftarrow (1-\alpha) Q(i,a) + \alpha [w_I(i,a,j) + w_R(i,a,j) \times$$

$$\frac{1 - \exp(-\tilde{\lambda} t(i,a,j))}{\tilde{\lambda}} + \exp(-\tilde{\lambda} t(i,a,j)) \max_{b \in \mathscr{A}(j)} Q(j,b)].$$

The convergence of the above can be worked out in a manner similar to that for the MDP above (Theorem 5). Note that as $\tilde{\lambda} \rightarrow 0$, the vanishing discount condition here, we should have no discounting; this is easily verified: as $\tilde{\lambda} \rightarrow 0$, $\frac{1 - \exp(-\tilde{\lambda} t(i,a,j))}{\tilde{\lambda}} \rightarrow t(i,a,j)$ and $\exp(-\tilde{\lambda} t(i,a,j)) \rightarrow 1$. For the risk-neutral case in [14], replace w by r, noting that they ignore the immediate reward.

A Two-time-Scale Algorithms: We now present two algorithms, one based on the SSP and the other based on a conditioning factor for the transition matrix. Conditioning factors are used commonly in policy gradient algorithms to ensure that a matrix is invertible [3, 40]. The SSP algorithm is based on the average reward algorithm in [27]; however, in [27] the slower iterate is assumed to start in the vicinity of its own optimal value, an assumption that we do not need here. For the SSP algorithm, after transition from i to j under a, update:

$$Q^{k+1}(i,a) \leftarrow (1-\alpha^k) Q^k(i,a) + \alpha^k [r(i,a,j) -$$

$$\theta v(i,a,j) - \phi^k t(i,a,j) + I_{\{j \neq i^*\}} \max_{b \in \mathscr{A}(j)} Q^k(j,b)] \quad (27)$$

where ϕ^k is the current estimate of the semi-variance-penalized long-run average reward, i^* is a special state chosen at the start, and $I_{\{\}}$ is an indicator function that equals 1 when the condition in the curly braces is met and is 0 otherwise. Here ϕ, and two other quantities, Γ and Ω, are updated in this transition only if a is a greedy

action; the updates are as follows:

$$\begin{aligned}\Omega^{k+1} &\leftarrow \Omega^k + r(i,a,j) - \theta v(i,a,j);\\ \Gamma^{k+1} &\leftarrow \Gamma^k + t(i,a,j);\\ \phi^k &\leftarrow (1-\beta^k)\phi^k + \beta^k(\Omega^k/\Gamma^k)\end{aligned} \tag{28}$$

in which step-sizes, α and β, are chosen such that $\lim_{k\to\infty}\beta^k/\alpha^k = 0$. We will initialize ϕ, Ω and Γ to 0. Also, α and β must satisfy the standard conditions for stochastic approximation.

The algorithm based on the conditioning factor, which helps produce a contraction, is as follows for the Q-factor:

$$Q^{k+1}(i,a) \leftarrow (1-\alpha^k)Q^k(i,a) + \alpha^k \times$$
$$\left[r(i,a,j) - \theta v(i,a,j) - \phi^k t(i,a,j) + \eta \max_{b\in\mathscr{A}(j)} Q^k(j,b)\right] \tag{29}$$

where $\eta \in (0,1)$ is the conditioning factor and the update on ϕ is as in (28). Note that if $t(i,a,j) = 1$ in the above, we obtain the MDP version of the conditioning factor algorithm, which is different from the vanishing discount algorithm (see Equation (26)) because of the presence of the term ϕ. This difference ensures that the conditioning factor can be quite small and still allow us to obtain optimal solutions.

We define $T : \Re^{|\mathscr{S}|\times|\mathscr{A}|\times 1} \to \Re^{|\mathscr{S}|\times|\mathscr{A}|}$:

$$(T(Q,\phi))(i,a) = \sum_{j=1}^{|\mathscr{S}|} p(i,a,j)[r(i,a,j) - \theta v(i,a,j) -$$
$$\phi t(i,a,j) + I_{\{j\neq i^*\}} \max_{j\in\mathscr{A}(i)} Q(j,b)].$$

and $T_\eta : \Re^{|\mathscr{S}|\times|\mathscr{A}|\times 1} \to \Re^{|\mathscr{S}|\times|\mathscr{A}|}$:

$$(T_\eta(Q,\phi))(i,a) = \sum_{j=1}^{|\mathscr{S}|} p(i,a,j)\times$$
$$\left[r(i,a,j) - \theta v(i,a,j) - \phi t(i,a,j) + \eta \max_{j\in\mathscr{A}(i)} Q(j,b)\right].$$

In general, we will denote the update of ϕ by:

$$\phi^{k+1} \leftarrow \phi^k + \beta^k(\Omega^k/\Gamma^k - \phi^k) = \phi^k + \beta^k(f(Q^k,\phi^k)), \tag{30}$$

where

$$f(Q^k,\phi^k) \equiv \Omega^k/\Gamma^k - \phi^k. \tag{31}$$

Our main equation in terms of T is:

$$Q(i,a) = (T(Q,\phi))(i,a) \ \ \forall (i,a). \tag{32}$$

and that for T_η is:

$$Q(i,a) = (T_\eta(Q,\phi))(i,a) \ \ \forall (i,a). \tag{33}$$

Assumption A1: The set of fixed points for the transformation $T(.,\phi)$ or $T_\eta(.,\phi)$ for fixed ϕ is non-empty. We will denote any fixed point in this set generally by $Q(\phi)$ for a given value of ϕ.

Assumption A2: $f : \Re^{|\mathscr{S}|\times|\mathscr{A}|} \to \Re$ is Lipschitz.

Assumption A3: $\frac{\partial f(Q,\phi)}{\partial \phi} < 0$.

Theorem 6: The RL algorithm in (27) and (28) converges to an optimal solution almost surely.

Proof: We first verify Assumption A1. Transformation $T(.,\phi)$ is contractive with respect to a weighted max norm for any ϕ (see [4]), and hence has a unique fixed point. For a fixed value of ϕ, that transformation $T(.,\phi)$ is non-expansive can be easily verified. Consider the O.D.E: $\frac{d\phi(t)}{dt} = 0$ and

$$\frac{dQ(t)}{dt} = T(Q(t),\phi(t)) - Q(t). \tag{34}$$

Under Assumption A1, from the result in [12], $Q(\phi)$ is an asymptotically stable equilibrium of the O.D.E in (34). It turns out that $\lim_{\zeta\to 0}\frac{T(\zeta Q,\phi)}{\zeta}$ exists since for every (i,a),

$$\begin{aligned}\lim_{\zeta\to 0}\frac{T(\zeta Q,\phi)(i,a)}{\zeta} &= \sum_{j=1}^{|\mathscr{S}|} p(i,a,j) I_{\{j\neq i^*\}} \max_{j\in\mathscr{A}(i)} Q(j,b)\\ &\equiv T_\infty(Q)(i,a).\end{aligned}$$

But from Assumption A1, the transformation $T_\infty(Q)$, which is a special case of $T(Q)$ with $w(.,.,.) \equiv 0$, must have at least one fixed point. Hence, the ODE

$$\frac{dQ(t)}{dt} = T_\infty(Q(t)) - Q(t)$$

has an asymptotically stable equilibrium. But the origin is the only fixed point here, and hence is the globally asymptotically stable equilibrium of (34). This implies from the result in [11] that Q^k remains bounded for a fixed ϕ. But the boundedness, along with Assumption A2, which follows from the definition of $f(.)$ in (31), the fact that T is Lipschitz continuous, and the conditions on the step sizes, implies from the two-time scale result in [9] that almost surely $Q^k \to Q(\phi^k)$. Then if $\delta^k = f(Q^k,\phi^k) - f(Q(\phi^k),\phi^k)$, a.s. $\delta^k \to 0$.

That ϕ is bounded is easily shown. Let

$$\frac{\max_{i,a,j}|w(i,a,j)|}{\min_{i,a,j} t(i,a,j)} = M < \infty,$$

where we assume that $t(i,a,j) > 0$ always. We can show that $|\phi^k| \le M$ for all k. Since $\phi^1 = 0$, we have:

$$|\phi^2| \le (1-\beta^k)|\phi^1| + \beta^k \frac{\max_{i,a,j}|w(i,a,j)|}{\min_{i,a,j} t(i,a,j)} = \beta M < M;$$

$$\begin{aligned}|\phi^{k+1}| &\le (1-\beta^k)|\phi^k| + \beta^k\left|\frac{\Omega^k}{\Gamma^k}\right|\\ &\le (1-\beta)M + \beta\left(\frac{k\max_{i,a,j}|w(i,a,j)|}{k\min_{i,a,j} t(i,a,j)}\right) = M.\end{aligned}$$

We now show that $\phi^k \to \phi^*$, where ϕ^* is the optimal variance-penalized average reward. Our arguments are based on those in [1]; however, since our update on the slower time scale is different, we need to work out the details. We define $\Delta^k = \phi^k - \phi^*$. Then, from the definition of δ^k and (30):

$$\Delta^{k+1} = \Delta^k + \beta^k f(Q(\phi^k), \phi^k) + \beta^k \delta^k. \qquad (35)$$

Using Assumption A3, which follows from the fact that $\frac{\partial f(Q,\phi)}{\partial \phi} = -1$ (see (31)), we have upper and lower bounds on the derivative, and hence there exist $L_1, L_2 \in \Re$ where $0 < L_1 \leq L_2$ such that:

$$\begin{aligned} -L_2(\phi_1 - \phi_2) \leq f(Q(\phi_1), \phi_1) - f(Q(\phi_2), \phi_2) \\ \leq -L_1(\phi_1 - \phi_2) \end{aligned}$$

for any scalar values of ϕ_1, ϕ_2. If the update in (27) is employed with $\phi^k \equiv \phi^*$, then because T will be non-expansive, from Assumption A1, using the result in [12], we will have that Q^k tends to a fixed point of T, but since the fixed point is a solution of the Q-factor version of the SSP Bellman equation in Lemma 1, $Q^k \to Q^*$. As a result, $(\Omega^k/\Gamma^k) \to \phi^*$ (since Ω and Γ are updated only under a greedy action), and from (30), $f(Q(\phi^*), \phi^*) = 0$. So if, $\phi_2 = \phi^*$ and $\phi_1 = \phi^k$, the above will lead to:

$$-L_2\Delta^k \leq f(Q(\phi^k), \phi^k) \leq -L_1\Delta^k.$$

Because $\beta^k > 0$, the above leads to:

$$-L_2\Delta^k\beta^k \leq \beta^k f(Q(\phi^k), \phi^k) \leq -L_1\Delta^k\beta^k.$$

The above combined with (35) leads to:

$$(1 - L_2\beta^k)\Delta^k + \beta^k\delta^k \leq \Delta^{k+1} \leq (1 - L_1\beta^k)\Delta^k + \beta^k\delta^k.$$

Then for any $\varepsilon > 0$, we have that:

$$\begin{aligned} (1 - L_2\beta^k)\Delta^k + \beta^k\delta^k - \varepsilon \leq \Delta^{k+1} \\ \leq (1 - L_1\beta^k)\Delta^k + \beta^k\delta^k + \varepsilon. \end{aligned}$$

Then the rest of the proof goes through as in Theorems 4.5 and 4.6 of [1] and we have that a.s., as $k \to \infty$, $\Delta^k \to 0$, i.e., $\phi^k \to \phi^*$, whence, $Q^k \to Q(\phi^*)$, but as argued above, $Q(\phi^*) = Q^*$. □

Theorem 7: The RL algorithm in (29) and (28) converges to an optimal solution almost surely.

Proof: Assumption A1 will be true because $T_\eta(.,\phi)$ is contractive in the max-norm, since $0 < \eta < 1$. The rest of the proof is similar to that above. However, the convergence will be to a solution of Equation (33), which technically will tend to the optimal solution as $\eta \to 1$. In practice, as we will show later, the conditioning factor can be quite small for obtaining optimal results. □

5. COMPUTATIONAL EXPERIMENTS

We consider a 2-state problem to illustrate the usefulness of the semi-variance metric. For μ, let R_μ denote the transition reward matrix and σ_μ^2 the variance. The values shown in Tables 1 and 2 are for the MDP and are obtained via exhaustive evaluation of the policies. They show that the variance-penalized optimal policy and the semi-variance-penalized optimal policy is not necessarily the same. In fact, the variance-penalized optimal policy is $(1,2)$, while the semi-variance-penalized optimal policy is $(2,1)$. In our experiments, we used $\theta = 0.15$, $\tau = 10$, and:

$$\mathbf{P}_{(1,1)} = \begin{bmatrix} 0.7 & 0.3 \\ 0.4 & 0.6 \end{bmatrix}; \mathbf{P}_{(2,2)} = \begin{bmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{bmatrix};$$

$$\mathbf{R}_{(1,1)} = \begin{bmatrix} 6.0 & -5 \\ 7.0 & 12 \end{bmatrix}; \mathbf{R}_{(2,2)} = \begin{bmatrix} 5.0 & 68 \\ -2 & 12 \end{bmatrix}.$$

Policy	ρ	σ^2	κ^2
(1,1)	5.8285	30.1420	46.5142
(1,2)	8.6250	31.2843	30.4750
(2,1)	11.0400	287.2384	18.7200
(2,2)	10.9500	187.5475	18.4500

Table 1: The average reward, variance, and semi-variance of different policies

Policy	$\rho - \theta\sigma^2$	$\rho - \theta\kappa^2$
(1,1)	1.3072	-1.14857
(1,2)	3.9323	4.0537
(2,1)	**-32.0457**	**8.2320**
(2,2)	-17.1821	8.1825

Table 2: Variance-penalized and semi-variance-penalized scores of the different polices. The semi-variance-penalized optimal solution is in bold.

The RVI algorithm converged to optimality in 12 iterations for $\varepsilon = 0.001$, while the vanishing-discount RL algorithm took no more than 100 iterations. We also conducted tests with a 2-state SMDP for we use the MDP above for the underlying Markov chains and the transition times in hours assume the following distributions. Under action 1: from state 1, unif(2,4) and from state 2, unif(3,5); under action 2, from state 1, unif(4,8) and from state 2, unif(6,10). We used $\theta = 0.15$ and $\tau = 1$ per hour. Exhaustive evaluation was done to determine the optimal policy which is action 2 in state 1 and action 1 in state 2. For the optimal policy, $\rho = 1.9714$, $\kappa^2 = 0.1429$, and $\phi^* = 1.95$. All three of the two-time-scale algorithms, the one based on DP and the the two

based on RL, converged to the optimal solution. Table 3 shows the Q-factors obtained from the three algorithms. The conditioning RL, the SSP-RL and the DP algorithms converged to optimality, producing $\phi = 1.88$, $\phi = 1.79$, $\phi = 1.942$, respectively. For DP, $\varepsilon = 0.01$; also for all algorithms, $\beta^k = 90/(100+k)$ and $\alpha^k = log(k)/k$. For the conditioning RL algorithm, $\eta = 0.99$; however, it produced optimal solutions for values as low as $\eta = 0.505$.

(i,a)	Conditioning RL	RL-SSP	DP
(1,1)	8.01	0.14	-4.4021
(1,2)	18.64	0.76	-0.0201
(2,1)	25.23	9.153	5.4670
(2,2)	14.73	9.145	-1.6106

Table 3: Q-factors from the DP and RL algorithms

6. CONCLUSIONS

We introduce a new risk measure, semi-variance, for MDPs and SMDPs in this paper. One of its notable advantages over variance is that it captures variability in a more accurate manner, while the associated control problem can still be solved, analogous to average reward, via DP. We proved that RVI (MDP case) for our problem converges in span; note that RVI, which is more stable than VI, has been shown to converge via other methods, but not in the span (in [20] via Lyapunov functions; [5] via Cauchy sequences; [42] via perturbation transformations). We extended a key SSP result for MDPs from [5], used widely in the literature (see e.g., [34]), to the SMDP case and presented a two-time-scale VI algorithm which exploits this result to solve the SMDP *without* discretization (conversion to MDP). For RL, we showed that eignenvalues of the transition matrix can be exploited for showing boundedness of a vanishing-discount procedure and also analyzed the undiscounted two-time-scale procedure.

Two avenues for future research are the study of semi-variance with policy gradients for semi-Markov control (see [17]; see also [16] for an excellent survey) and hierarchical control of semi-Markov processes (see [33]). Both [17, 33] present some path-breaking results in the field of semi-Markov control.

Acknowledgements: The author would like to acknowledge support from NSF grant ECCS: 0841055 that partially funded this research.

APPENDIX A

Def 3. Let $\mathbf{A}$ be a matrix with W rows and C columns with non-negative elements; additionally, let the elements in each row of this matrix sum to 1. Let $\mathscr{W} = \{1,2,\ldots,W\}$ and $\mathscr{C} = \{1,2,\ldots,C\}$. Now let $b(i,j,l) = \min_{l\in\mathscr{C}}\{A(i,l),A(j,l)\}$ for every $(i,j) \in \mathscr{W}\times\mathscr{W}$, where $A(i,j)$ denotes the element of the ith row and the jth column in $\mathbf{A}$. Further let $B(i,j) = \sum_{l=1}^{C} b(i,j,l)$ for every $(i,j) \in \mathscr{W}\times\mathscr{W}$. The delta-coefficient, α, of a matrix, $\mathbf{A}$, is then defined as:

$$\alpha_{\mathbf{A}} = 1 - \min_{(i,j)\in\mathscr{W}\times\mathscr{W}} B(i,j). \quad \text{(A,1)}$$

Lemma 2: Let $\vec{x}$ be any arbitrary column vector with C components and $\mathbf{A}$ be a matrix with C columns, where C is finite. Then, $sp(\mathbf{A}\vec{x}) \leq \alpha_{\mathbf{A}}\ sp(\vec{x})$.

For a proof of the above lemma, see Seneta [39]. The following notational conventions will be adopted in what follows: $L^{k+1}\vec{z} \equiv L\left(L^k\vec{z}\right)$, and a vector $\vec{z}^{\,k}$ transformed m times will be referred to as $\vec{z}^{\,k+m}$. Also, for any $i \in \mathscr{S}$,

$$d_{x^k}(i) \in \arg\max_{a\in\mathscr{A}(i)} \left[\bar{w}(i,a) + \sum_{j\in\mathscr{S}} p(i,a,j)x^k(j)\right]. \quad \text{(A,2)}$$

Thus d_{x^k} will denote a policy that will prescribe the action defined in (A,2) for the ith state. If $\vec{y}^{\,k}$ is a vector,

$$L_{d_{x^k}}y^k(i) = \left[\bar{w}(i,d_{x^k}(i)) + \sum_{j\in\mathscr{S}} p(i,d_{x^k}(i),j)y^k(j)\right]$$

for every $i \in \mathscr{S}$. Thus, for every $i \in \mathscr{S}$ and any vector $\vec{x}^{\,k}$,

$$Lx^k(i) \equiv L_{d_{x^k}}x^k(i). \quad \text{(A,3)}$$

Lemma 3: Let L denote the Bellman optimality operator defined in (2) and M be a positive finite integer. Consider two vectors $\vec{x}^{\,1}$ and $\vec{y}^{\,1}$ that have $|\mathscr{S}|$ components. Also, using $\mathbf{P}_\mu$ to denote the transition probability matrix associated with policy μ, we define the following matrices:

$$\mathbf{A}_x^M \equiv \mathbf{P}_{d_{x^M}}\mathbf{P}_{d_{x^{M-1}}}\ldots\mathbf{P}_{d_{x^1}} \text{ and } \mathbf{A}_y^M \equiv \mathbf{P}_{d_{y^M}}\mathbf{P}_{d_{y^{M-1}}}\ldots\mathbf{P}_{d_{y^1}}.$$

Then $sp(L^M\vec{y}^{\,1} - L^M\vec{x}^{\,1}) \leq \alpha_{\mathbf{A}}\ sp(\vec{y}^{\,1} - \vec{x}^{\,1})$, where $\mathbf{A} \equiv \left[\frac{\mathbf{A}_y^M}{\mathbf{A}_x^M}\right]$.

Proof: Let states s^* and s_* be defined as follows:

$$s^* = \arg\max_{s\in\mathscr{S}}\{L^My^1(s) - L^Mx^1(s)\};$$

$$s_* = \arg\min_{s\in\mathscr{S}}\{L^My^1(s) - L^Mx^1(s)\}.$$

$$\text{For any } i \in \mathscr{S}, L^Mx^1(i) = L_{d_{x^M}}L_{d_{x^{M-1}}}\ldots L_{d_{x^2}}L_{d_{x^1}}x^1(i). \quad \text{(A,4)}$$

We can show that

$$L^My^1(i) \geq L_{d_{x^M}}L_{d_{x^{M-1}}}\ldots L_{d_{x^2}}L_{d_{x^1}}y^1(i)\forall i. \quad \text{(A,5)}$$

The above can be proved as follows. From definition, for all $i \in \mathscr{S}$, $Ly^1(i) \geq L_{d_{x^1}} y^1(i)$. Since L is monotonic (from Lemma 1.1.1 of [5] (Vol 2, pp 7) via replacing $\bar{r}$ by $\bar{w}$), for all $i \in \mathscr{S}$, $L\left(Ly^1(i)\right) \geq L\left(L_{d_{x^1}} y^1(i)\right)$. From definition of L in (Def 2.), for all $i \in \mathscr{S}$, $L\left(Ly^1(i)\right) \geq L_{d_{x^2}}\left(L_{d_{x^1}} y^1(i)\right)$. From the preceding inequalities, for all $i \in \mathscr{S}$, $L\left(Ly^1(i)\right) \geq L_{d_{x^2}}\left(L_{d_{x^1}} y^1(i)\right)$. In this way, by repeatedly using the monotonicity property, we can establish (A,5). From (A,4) and (A,5), it follows that

$$
\begin{aligned}
&L^M y^1(s^*) - L^M x^1(s^*) \\
&\geq [L_{d_{x^M}} L_{d_{x^{M-1}}} \ldots L_{d_{x^1}} y^1(s^*)] - [L_{d_{x^M}} L_{d_{x^{M-1}}} \ldots L_{d_{x^1}} x^1(s^*)] \\
&= [\bar{w}(s^*, d_{x^1}(s^*)) + \bar{w}(s^*, d_{x^2}(s^*)) + \cdots + \bar{w}(s^*, d_{x^M}(s^*)) + \\
&\mathbf{P}_{d_{x^M}} \mathbf{P}_{d_{x^{M-1}}} \ldots \mathbf{P}_{d_{x^1}} y^1(s^*)] - \\
&[\bar{w}(s^*, d_{x^1}(s^*)) + \bar{w}(s^*, d_{x^2}(s^*)) + \cdots + \bar{w}(s^*, d_{x^M}(s^*)) + \\
&\mathbf{P}_{d_{x^M}} \mathbf{P}_{d_{x^{M-1}}} \ldots \mathbf{P}_{d_{x^1}} x^1(s^*)] = \mathbf{A}_x^M (y^1 - x^1)(s^*).
\end{aligned}
$$

Thus:

$$
L^M y^1(s^*) - L^M x^1(s^*) \leq \mathbf{A}_x^M (y^1 - x^1)(s^*). \qquad \text{(A,6)}
$$

Using logic similar to that used above:

$$
L^M y^{\,1}(s_*) - L^M x^{\,1}(s_*) \leq \mathbf{A}_y^M (y^1 - x^1)(s_*). \qquad \text{(A,7)}
$$

Then,

$$
\begin{aligned}
&sp(L^M \vec{y}^{\,1} - L^M \vec{x}^{\,1}) \\
&= \{L^M y^1(s^*) - L^M x^1(s^*)\} - \{L^M y^1(s_*) - L^M x^1(s_*)\} \\
&\leq \mathbf{A}_y^M (y^1 - x^1)(s^*) - \mathbf{A}_x^M (y^1 - x^1)(s_*) \\
&\text{(from (A,6) and (A,7))} \\
&\leq \max_{s \in \mathscr{S}} \mathbf{A}_y^M (y^1 - x^1)(s) - \min_{s \in \mathscr{S}} \mathbf{A}_x^M (y^1 - x^1)(s) \\
&\leq \max_{s \in \mathscr{S}} \left[\frac{\mathbf{A}_y^M}{\mathbf{A}_x^M}\right] (y^1 - x^1)(s) - \min_{s \in \mathscr{S}} \left[\frac{\mathbf{A}_y^M}{\mathbf{A}_x^M}\right] (y^1 - x^1)(s) \\
&= sp\left(\left[\frac{\mathbf{A}_y^M}{\mathbf{A}_x^M}\right] (\vec{y}^{\,1} - \vec{x}^{\,1})\right) \\
&\leq \alpha_{\mathbf{A}}\, sp(\vec{y}^{\,1} - \vec{x}^{\,1}) \text{ (from Lemma 2).}
\end{aligned}
$$

□

REFERENCES

[1] J. Abounadi, D. Bertsekas, and V. Borkar. Learning algorithms for Markov decision processes with average cost. *SIAM Journal of Control and Optimization*, vol. 40, pp. 681-698, 2001.

[2] E. Altman. *Constrained Markov decision processes*. CRC Press, Boca Raton, 1998.

[3] J. Baxter and P. Bartlett. Infinite-horizon policy-gradient estimation. *Journal of Artificial Intelligence*, vol. 15, pp. 319-350, 2001.

[4] D.P. Bertsekas and J. Tsitsiklis. *Neuro-Dynamic Programming*. Athena, Belmont, 1996.

[5] D.P. Bertsekas. *Dynamic Programming and Optimal Control*. Athena, Belmont, 2nd edition, 2000.

[6] T. Bielecki, D. Hernandez-Hernandez, and S. Pliska. Risk-sensitive control of finite state Markov chains in discrete time.*Math. Methods of Opns. Research*, vol. 50, pp. 167-188, 1999.

[7] K. Boda and J. Filar. Time consistent dynamic risk measures. *Mathematical Methods of Operations Research*, vol. 63, pp. 169–186, 2005.

[8] V. Borkar and S. Meyn. Risk-sensitive optimal control for Markov decision processes with monotone cost. *Mathematics of Operations Research*, vol. 27, pp. 192-209, 2002.

[9] V. S. Borkar. Stochastic approximation with two-time scales. *Systems and Control Letters*, vol. 29, pp. 291-294, 1997.

[10] V. S. Borkar. Asynchronous stochastic approximation. *SIAM Journal of Control and Optimization*, vol. 36, no. 3, pp. 840-851, 1998.

[11] V. S. Borkar and S.P. Meyn. The ODE method for convergence of stochastic approximation and reinforcement learning. *SIAM Journal of Control and Optimization*, vol. 38, no. 2, pp. 447-469, 2000.

[12] V.S. Borkar and K. Soumyanath. A new analog parallel scheme for fixed point computation, part I: Theory. *IEEE Transactions on Circuits and Systems I: Theory and Applications*, vol. 44, pp. 351-355, 1997.

[13] M. Bouakiz and Y. Kebir. Target-level criterion in Markov decision processes. *Journal of Optimization Theory and Applications*, vol. 86, pp. 1-15, 1995.

[14] S.J. Bradtke and M. Duff. Reinforcement learning methods for continuous-time MDPs. In *Advances in Neural Information Processing Systems 7*. MIT Press, Cambridge, MA, USA, 1995.

[15] F. Brauer and J. Nohel. *The qualitative Theory of Ordinary Differential Equations: An Introduction*. Dover Publishers, New York, 1989.

[16] Xi-Ren Cao. From perturbation analysis to Markov decision processes and reinforcement learning. *Discrete-Event Dynamic Systems: Theory and Applications*, vol. 13, pp. 9-39, 2003.

[17] Xi-Ren Cao. Semi-Markov decision problems and performance sensitivity analysis. *IEEE Transactions on Automatic Control*, vol. 48, no. 5, pp. 758-768, 2003.

[18] R. Cavazos-Cadena. Solution to risk-sensitive average cost optimality equation in a class of MDPs with finite state space. *Math. Methods of Opns. Research*, vol. 57, pp. 253–285, 2003.

[19] R. Cavazos-Cadena and E. Fernandez-Gaucherand. Controlled Markov chains with risk-sensitive criteria. *Mathematical Models of Operations Research*, vol. 43, pp. 121–139, 1999.

[20] R-R. Chen and S Meyn. Value iteration and optimization of multiclass queueing networks. *Queueing Systems*, vol. 32, pp. 65-97, 1999.

[21] K. Chung and M. Sobel. Discounted MDPs: Distribution functions and exponential utlity maximization. *SIAM Journal of Control and Optimization*, vol. 25, pp. 49-62, 1987.

[22] G. Di Masi and L. Stettner. Risk-sensitive control of discrete-time Markov processes with infinite horizon. *SIAM Journal of Control and Optimization*, vol. 38, no. 1, pp. 61–78, 1999.

[23] J. Estrada. Mean-semivariance behavior: Downside risk and capital asset pricing. *International Review of Economics and Finance*, vol. 16, pp. 169-185, 2007.

[24] J. Filar, L. Kallenberg, and H. Lee. Variance-penalized Markov decision processes. *Mathematics of Operations Research*, vol. 14, no 1, pp. 147–161, 1989.

[25] J. Filar, D. Krass, and K. Ross. Percentile performance criteria for limiting average Markov decision processes. *IEEE Transactions on Automatic Control*, vol. 40, pp. 2-10, 1995.

[26] W. Fleming and D. Hernandez-Hernandez. Risk-sensitive control of finite state machines on an infinte horizon. *SIAM Journal of Control and Optimization*, vol. 35, pp. 1790–1810, 1997.

[27] A. Gosavi. Reinforcement learning for long-run average cost. *European Journal of Operational Research*, vol. 155, pp. 654–674, 2004.

[28] A. Gosavi. A risk-sensitive approach to total productive maintenance. *Automatica*, vol. 42, pp. 1321–1330, 2006.

[29] A. Gosavi, S. L. Murray, V. M. Tirumalasetty, and S. Shewade. A budget-sensitive approach to total productive maintenance. To appear in *Engineering Management Journal*, 2011.

[30] D. Hernandez-Hernandez and S. Marcus. Risk-sensitive control of Markov processes in countable state space. *Systems and Control Letters*, vol. 29, pp. 147–155, 1996.

[31] R. Howard and J. Matheson. Risk-sensitive MDPs. *Management Science*, vol. 18, no. 7, pp. 356–369, 1972.

[32] G. Hübner. Improved precedures for eliminating sub-optimal actions in Markov programming by the use of contraction properties. In *Transactions of 7th Prague Conference, 1974*. Dordrecht, 1978.

[33] Qi Jiang, H-S. Xi, and B-Q. Yin. Dynamic file grouping for load balancing in streaming media clustered server systems. *International Journal of Control, Automation, and Systems*, vol. 7, no. 4, pp. 630–637, 2009.

[34] W. Y. Kwon, H. Suh, and S. Lee. SSPQL: Stochastic Shortest Path-based Q-learning, *International Journal of Control, Automation, and Systems*, vol. 9, no. 2, pp.328-338, 2011

[35] A. E. B. Lim and X.Y. Zhou. Risk-sensitive control with HARA utility. *IEEE Transactions on Automatic Control*, vol. 46, no. 4, pp.563–578, 2001.

[36] R. Porter. Semivariance and stochastic dominance. *American Economic Review*, vol. 64, pp. 200–204, 1974.

[37] M. L. Puterman. *Markov Decision Processes*. Wiley Interscience, New York, 1994.

[38] S. Ross. *Applied Probability Models with Optimization Applications*. Dover, New York, 1992.

[39] E. Seneta. *Non-Negative Matrices and Markov Chains*. Springer-Verlag, NY, 1981.

[40] S. Singh, V. Tadic, and A. Doucet. A policy-gradient method for semi-Markov decision processes with application to call admission control. *European Journal of Operational Research*, vol. 178, no. 3, pp. 808–818, 2007.

[41] M. Sobel. The variance of discounted Markov decision processes. *Journal of Applied Probability*, vol. 19, pp. 794–802, 1982.

[42] H. C. Tijms. *A first course in stochastic models*. UK, NY, second edition, 2003.

[43] C.G. Turvey and G. Nayak. The semi-variance-minimizing hedge ratios. *Journal of Aggrictultural and Resource Economics*, vol. 28, no. 1, pp. 100–115, 2003.

[44] D. White. Minimizing a threshold probability in discounted Markov decision processes. *Journal of Mathematical Analysis and Applications*, vol. 173, pp. 634–646, 1993.

[45] C. Wu and Y. Lin. Minimizing risk models in Markov decision processes with policies depending on target values. *Journal of Mathematical Analysis and Applications*, vol. 231, pp. 47–67, 1999.

Abhijit Gosavi received the B.E in Mechanical Engineering from Jadavpur University in 1992, an M.Tech in Mechanical Engineering from the Indian Institute of Technology, Madras in 1995, and a Ph.D. in Industrial Engineering from the University of South Florida. His research interests include Markov decision processes, simulation, and applied operations research. He joined the Missouri University of Science and Technology in 2008 as an Assistant Professor.