Journal of Artificial Intelligence Research 75 (2022) 569-595 Submitted 04/2022; published 10/2022

Mean-Semivariance Policy Optimization via Risk-Averse Reinforcement Learning

Xiaoteng Ma MA-XT17@MAILS.TSINGHUA.EDU.CN
Department of Automation, Tsinghua University,
Beijing, 100086, P. R. China

Shuai Ma MASH35@MAIL.SYSU.EDU.CN
School of Business, Sun Yat-sen University,
Guangzhou, 510275, P. R. China

Li Xia XIALI5@SYSU.EDU.CN
(Corresponding author)
School of Business, Sun Yat-sen University,
Guangzhou, 510275, P. R. China

Qianchuan Zhao ZHAOQC@TSINGHUA.EDU.CN
Department of Automation, Tsinghua University,
Beijing, 100086, P. R. China

Abstract

Keeping risk under control is often more crucial than maximizing expected reward in real-world decision-making situations, such as finance, robotics, autonomous driving, etc. The most natural choice of risk measures is variance, while it penalizes the upside volatility as much as the downside part. Instead, the (downside) semivariance, which captures negative deviation of a random variable under its mean, is more suitable for risk-averse proposes. This paper aims at optimizing the mean-semivariance (MSV) criterion in reinforcement learning w.r.t. steady reward distribution. Since semivariance is time-inconsistent and does not satisfy the standard Bellman equation, the traditional dynamic programming methods are inapplicable to MSV problems directly. To tackle this challenge, we resort to Perturbation Analysis (PA) theory and establish the performance difference formula for MSV. We reveal that the MSV problem can be solved by iteratively solving a sequence of RL problems with a policy-dependent reward function. Further, we propose two on-policy algorithms based on the policy gradient theory and the trust region method. Finally, we conduct diverse experiments from simple bandit problems to continuous control tasks in MuJoCo, which demonstrate the effectiveness of our proposed methods.

1. Introduction

Reinforcement learning (RL) has shown great promise in solving complex decision problems, such as Go (Silver et al., 2017), video games (Berner et al., 2019; Vinyals et al., 2019) and dexterous robotic control (Nagabandi et al., 2020). Learning by trial and error, RL enables an agent to maximize its accumulated expected rewards through the interaction with a simulator. However, RL deployment in real-world scenarios is still challenging and unreliable (Garcıa & Fernández, 2015; Dulac-Arnold et al., 2019). One of the reasons is that real decision-makers need to consider multi-objective functions. The desired policy

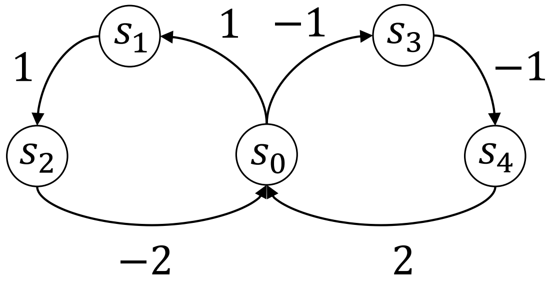


Figure 1: A toy example illustrates the effect of MSV. We refer the policy going left as l and the other as r. Two policies have the same average return $\eta^l = \eta^r = 0$ and the same variance $\zeta^l = \zeta^r = 2$. However, since the semivariance $\zeta^l_- = 4/3 > \zeta^r_- = 2/3$, the policy going right has a smaller (downside) semivariance. It shows that MSV enables to avoid extreme costs compared with MV.

should perform well for broader metrics, not just for expectation. That raises the demand of *risk-sensitive learning*, which aims at balancing the return and risk in face of uncertainty.

The risk-sensitive decision-making has been widely studied beyond the scope of RL, which can be traced back to the *mean-variance* (MV) optimization theory established by Markowitz (Markowitz, 1952). Variance, which captures the fluctuation and concentration of random variables, is a natural choice of the risk measure. As Markowitz only considers the single-period problem, many studies focus on extending the results to multi-period scenarios, from stochastic control (Li & Ng, 2000) to Markov decision process (Sobel, 1982; Filar et al., 1989). However, the variance of a multi-period problem depends on the average value of the whole process. It breaks the essential property of dynamic programming—time-consistency, and makes it hard to design model-free learning algorithms under the standard RL framework. Developing an efficient algorithm to optimize MV is still an ongoing topic in the RL community (Xie et al., 2018; Bisi et al., 2020; Xia, 2020; Zhang et al., 2021; Ma et al., 2022b, 2022a).

While MV analysis is the most widely applied risk-return analysis in practice, variance metric is questionable as a risk measure. As a measure of volatility, variance penalizes upside deviations from the mean as much as downside deviations. It could be problematic as the upside deviation comes from the higher return which is desirable. In general, the outcome distributions in the real world are often asymmetrical, such as the ones in the stock market (Estrada, 2007; Bollerslev et al., 2020), suggesting that we should control the "good" and "bad" volatility separately. Hence, Markowitz (1959) presents the *mean-semivariance* (MSV) as an alternative measure, which only penalizes the "bad" volatility, performing as a downside risk indicator. Even if the distribution is symmetrical, optimizing MSV is at least effective as optimizing MV. To better illustrate the difference between variance and semivariance, we construct a simple MDP example shown in Figure 1. The two policies result in two reward distributions symmetrically, for which variances are indistinguishable. However, the policy going right is preferred since it results in a lower semivariance.

Though MSV is a more plausible measure of risk, optimizing MSV is even more complicated than MV. It inherits time-inconsistency from variance and introduces a truncation

Algorithm 1 The framework of MSV optimization

```
Initialize policy as $\mu$
repeat
    Evaluate $\mu$ and get $\eta$ (cf. Equation 1) and $\eta_-$ (cf. Equation 12)
    Set reward function as $g = (1 + 2\beta\eta_-)r - \beta(r - \eta)_-^2$
    $\mu \leftarrow$ POLICY_UPDATE($\mu, g$)
until $\mu$ converges
```

function of mean, making the analysis non-trivial. Due to the complexity of this objective, existing works consider a subset of problems restricted with a fixed mean (Wei, 2019) or heuristic algorithms for MSV (Yan et al., 2007; Zhang et al., 2012; Liu & Zhang, 2015; Chen et al., 2019). To the best of our knowledge, there are currently no relevant studies on MSV in the RL literature.

In this paper, we aim to fill the gap of the previous study on the single-period MSV problem and extend the static methods to online RL algorithms. To achieve that, we resort to Perturbation Analysis (PA) theory (Cao, 2007) (also called the sensitivity-based optimization theory or the relative optimization theory) for Markov systems, which lays the basis of many efficient RL methods, such as TRPO (Schulman et al., 2015), CPO (Achiam et al., 2017) and MBPO (Janner et al., 2019). The contributions of our work are threefold. *Firstly*, instead of constructing a Bellman operator, we establish the MSV performance difference formula of two policies (see Section 4 for details). The result indicates that the performance difference can be decomposed into two parts: the improvement corresponding to a reward function depending on the current policy and the average performance change from the current to the updated one. *Second*, we iteratively optimize MSV by considering the shift in mean locally and constructing a surrogate reward function. The framework is shown in Algorithm 1. Under this framework, we develop two algorithms based on the policy gradient theory and the trust region method, respectively. We show that optimizing the surrogate reward function in the trust region has a similar performance lower bound with the standard TRPO, which guarantees the monotonic improvement if the trust region is tight. *Finally*, we conduct diverse experiments to examine the effectiveness of our proposed methods, including a bandit problem, a tabular portfolio management problem, and robotic control tasks based on MuJoCo. The results demonstrate that the proposed algorithms successfully improve the performance under the criterion of MSV, which is better than standard RL from a risk-averse perspective.

2. Related Work

Below we briefly review the literature about optimization of MSV and other risk measures.

2.1 Mean-Semivariance

MSV is first introduced by Markowitz (1959) as an alternative to MV. Thereafter, many researchers study portfolio selection problems by employing the semivariance as the risk measure (Markowitz et al., 1993; Hogan & Warren, 1974; Choobineh & Branting, 1986; Briec & Kerstens, 2009), most of which are limited to the single-period problem. Due to the complexity of MSV, previous studies on MSV in multi-period problems resort to heuristic methods, such as fuzzy systems and genetic algorithms (Yan et al., 2007; Zhang et al., 2012; Liu & Zhang, 2015; Chen et al., 2019). Wei (2019) studies a special case of MSV in the continuous-time MDP, where the mean of the discounted total cost is equal to a given function. Another stream of researches (Tamar et al., 2016; Shapiro et al., 2021) study semideviation instead of semivariance. As standard deviation is an alternative to variance, semideviation is considered as an alternative to semivariance. The main benefit of mean-semideviation (MSD) is that it satisfies the property "coherent," and hence it can be written in a Bellman form (Ruszczyński, 2010). However, the additional square operation makes optimizing MSD with a data-driven approach non-trivial. We leave the optimization of MSD in RL as future work. Furthermore, maximizing the upside semivariance could improve the exploration ability (Mavrin et al., 2019; Ma et al., 2020; Zhou et al., 2020), showing the potential of MSV from an opposite perspective.

2.2 Mean-Variance

Since MSV is highly related to MV, in this part, we summarize the works on MV in Markov decision processes (MDPs) and RL. Based on the definition of variance in the framework of MDPs, the existing studies on variance can be broadly divided into two categories. One stream of works (Sobel, 1982; Castro et al., 2012; Prashanth & Ghavamzadeh, 2016; Xie et al., 2018) concern the variance of total return $R = \sum_{t=0}^{\infty} \gamma^t r_t$ under the initial state distribution, i.e., $\mathbb{V}_{\pi_0}(R)$ where γ is the discount factor, π_0 is the initial state distribution and r_t is the reward at the stage t. This definition concerns the risk of total rewards at the final stage, while we are more concerned about long-term volatility in practical problems. Hence, the long-run variance (Filar et al., 1989; Chung, 1994; Gosavi, 2014; Xia, 2016, 2020; Bisi et al., 2020; Zhang et al., 2021), also known as the steady-state variance, is proposed to describe the variance of the steady reward distribution. The long-run variance is defined by $\lim_{T\to\infty} \frac{1}{T}\mathbb{E}_{\pi_0,\mu}\left[\sum_{t=0}^{T-1}\left(r(s_t,a_t)-\eta^{\mu}\right)^2\right]$ (cf. Equation 3), where η^{μ} is the long-run average of policy μ. Since the average reward η^{μ} depends on the current policy, it breaks the time-consistency. To handle this problem, Xia (2016, 2020) derives a variance performance difference formula with PA and proposes a policy iteration algorithm which is guaranteed to converge to a local optimum. In this paper, we adopt similar definition of Xia's work and extend the formulation from MV to MSV.

2.3 Other Risk Measures

Besides the MV and MSV, other risk measures capture different features of the return distribution. A classical risk measure in optimal control is exponential utility (Howard & Matheson, 1972; Borkar & Meyn, 2002; Fei et al., 2020). The exponential utility enjoys a product form of the Bellman equation. Therefore the corresponding value-based algorithms such as Q-learning are well-developed. While the exponential Bellman equation is elegant in theory, it poses some computational problems as the exponential values are often too large to be numerically calculated. Another famous risk measure is Conditional Value at Risk (CVaR), defined as the average value under the α-quantile. Many existing methods (Nemirovski & Shapiro, 2007; Chow & Ghavamzadeh, 2014; Tamar et al., 2015; Chow et al., 2015, 2017) optimize CVaR as the objective or constraints. The main difference between CVaR and

MSV is that CVaR puts even weights for the events under a certain threshold, while the importance of the extreme values on the concerned side increases quadratically in MSV. We refer to Delage et al.'s work (2019) for more discussion on the connection of different risk measures.

3. Preliminaries

In this paper, we focus on the infinite-horizon discrete-time MDP as $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, r, P, \pi_0 \rangle$, where $\mathcal{S}$ denotes the state space, $\mathcal{A}$ denotes the action space, $r : \mathcal{S} \times \mathcal{A} \mapsto [-R_{\max}, R_{\max}]$ denotes a bounded reward function and $P : \mathcal{S} \times \mathcal{A} \mapsto \Delta(\mathcal{S})$ is the transition matrix and $\pi_0 \in \Delta(\mathcal{S})$ denotes the initial state distribution. We assume that all the involved MDPs are ergodic. Let $\mu : \mathcal{S} \mapsto \Delta(\mathcal{A})$ denote a Markovian randomized policy and Π denote the randomized policy space.

We are interested in the long-run average reward

$$\eta^\mu := \lim_{T\to\infty} \frac{1}{T} \mathbb{E}_{\pi_0,\mu} \left[\sum_{t=0}^{T-1} r(s_t, a_t) \right], \tag{1}$$

where $\mathbb{E}_{\pi_0,\mu}$ stands for the expectation with $s_0 \sim \pi_0, a_t \sim \mu(\cdot \mid s_t), s_{t+1} \sim P(\cdot \mid s_t, a_t)$. Note that η^μ is independent of π_0 when $T \to \infty$. With π denoting the steady state distribution, it is convenient to rephrase the long-run average reward as

$$\eta^\mu := \mathbb{E}_{s\sim\pi, a\sim\mu} \left[r(s,a) \right]. \tag{2}$$

The variance and semivariance w.r.t. μ are defined by

$$\zeta^\mu := \lim_{T\to\infty} \frac{1}{T} \mathbb{E}_{\pi_0,\mu} \left[\sum_{t=0}^{T-1} \left(r(s_t, a_t) - \eta^\mu\right)^2 \right], \tag{3}$$

$$\zeta_-^\mu := \lim_{T\to\infty} \frac{1}{T} \mathbb{E}_{\pi_0,\mu} \left[\sum_{t=0}^{T-1} \left(r(s_t, a_t) - \eta^\mu\right)_-^2 \right], \tag{4}$$

where $(\cdot)_- := \min\{0, \cdot\}$. In this paper, we focus on the mean-semivariance criterion,

$$\xi_-^\mu := \eta^\mu - \beta \zeta_-^\mu,$$

where $\beta \geq 0$ is the parameter for the trade-off between mean and semivariance. Analogously, when mean-variance criterion is mentioned, we mean $\xi^\mu := \eta^\mu - \beta\zeta^\mu$.

We further respectively define state-value function, action-value function, and advantage function for average reward as

$$\begin{aligned} V_\eta^\mu(s) &:= \mathbb{E}_\mu \left[\sum_{t=0}^{\infty} (r(s_t, a_t) - \eta^\mu) \mid s_0 = s \right], \\ Q_\eta^\mu(s,a) &:= \mathbb{E}_\mu \left[\sum_{t=0}^{\infty} (r(s_t, a_t) - \eta^\mu) \mid s_0 = s, a_0 = a \right], \\ A_\eta^\mu(s,a) &:= Q_\eta^\mu(s,a) - V_\eta^\mu(s). \end{aligned}$$

Similarly, the value functions for semivariance are defined as

$$\begin{aligned}
V^{\mu}_{\zeta_-}(s) &:= \mathbb{E}_{\mu}\left[\sum_{t=0}^{\infty}\left((r(s_t,a_t)-\eta^{\mu})^2_- - \zeta^{\mu}_-\right) \mid s_0 = s\right], \\
Q^{\mu}_{\zeta_-}(s,a) &:= \mathbb{E}_{\mu}\left[\sum_{t=0}^{\infty}\left((r(s_t,a_t)-\eta^{\mu})^2_- - \zeta^{\mu}_-\right) \mid s_0 = s, a_0 = a\right], \\
A^{\mu}_{\zeta_-}(s,a) &:= Q^{\mu}_{\zeta_-}(s,a) - V^{\mu}_{\zeta_-}(s).
\end{aligned}$$

For notation simplicity, we will omit the superscript "μ" when the context is clear, e.g., the average rewards $\eta^{\mu}, \eta^{\mu'}$ are written as η, η' instead. When r is mentioned, we omit (s,a) and use r in short.

Before our analysis of MSV, we briefly review the average-reward policy gradient theorem and the trust region theorem.

Theorem 1 (Average-Reward Policy Gradient by Sutton & Barto, 2018)**.** *For a policy μ parameterized by θ, we have*

$$\nabla_{\theta}\eta = \mathbb{E}_{s\sim\pi,a\sim\mu}[\nabla_{\theta}\log\mu(a \mid s)A^{\mu}_{\eta}(s,a)].$$

Theorem 2 (Average-Reward Trust Region Policy Optimization by Zhang & Ross, 2021; Ma et al., 2021)**.** *Consider the following problem,*

$$\begin{aligned}
&\max_{\mu_{\theta}} \mathcal{L}^{\mu}(\mu_{\theta}), \qquad (5)\\
&\text{s.t. } \mathbb{E}_{s\sim\pi}D_{\mathrm{TV}}(\mu_{\theta}(\cdot \mid s) \parallel \mu(\cdot \mid s)) \le \epsilon_{\mu},
\end{aligned}$$

where

$$\mathcal{L}^{\mu}(\mu_{\theta}) := \mathbb{E}_{s\sim\pi,a\sim\mu_{\theta}}\left[A^{\mu}_{\eta}(s,a)\right]. \qquad (6)$$

Denote μ' as the solution of the above problem. The following bound holds:

$$\eta' - \eta \ge \mathcal{L}^{\mu}(\mu') - 2(\kappa' - 1)\epsilon_{\eta}\epsilon_{\mu}, \qquad (7)$$

where $\epsilon_{\eta} = \max_s |\mathbb{E}_{a\sim\mu'}[A^{\mu}_{\eta}(s,a)]|$ and κ' is Kemeny's constant under μ'.

4. Perturbation Analysis

In this section, we derive the *MSV performance difference formula* (MSVPDF), where the core concept—performance difference formula—comes from the PA for Markov systems, also called the sensitivity-based optimization theory. With the aid of MSVPDF, we obtain the necessary optimality condition for the MSV problem. It also lays the basis for developing optimization algorithms (see Section 5), such as the policy gradient method and the trust region method. For readers unfamiliar with PA, we provide a brief review of the theory in Appendix A.

4.1 Performance Difference Formula

MSVPDF is formally stated as below.

Theorem 3. *For any two policies $\mu, \mu' \in \Pi$, we have*

$$\xi'_- - \xi_- = \mathbb{E}_{s\sim\pi', a\sim\mu'}[A^\mu_\eta(s,a) - \beta A^\mu_{\zeta_-}(s,a)] - \beta\mathbb{E}_{s\sim\pi', a\sim\mu'}[(r-\eta')^2_- - (r-\eta)^2_-] \quad (8)$$

Proof. To decompose the policy performance with the policy-dependent reward, we first introduce a pseudo mean λ. We analyze the policy difference with the pseudo mean and corresponding pseudo reward function, and then turn into the true mean by letting $\lambda = \eta$.

With a pseudo mean λ, we transform the original problem into a standard MDP with reward function

$$f(s,a) := r - \beta(r-\lambda)^2_-. \quad (9)$$

We obtain a pseudo mean-semivariance objective by optimizing this pseudo reward-function,

$$\xi_{\lambda,-} := \xi^\mu_{\lambda,-} = \mathbb{E}_{s\sim\pi, a\sim\mu}\left[f(s,a)\right].$$

By definition, we have

$$\xi_- - \xi_{\lambda,-} = \mathbb{E}_{s\sim\pi, a\sim\mu}\left[r - \beta(r-\eta)^2_- - f(s,a)\right].$$

Since the pseudo reward is independent of the policy, we can write its performance difference formula directly (Cao, 2007, Chapter 2):

$$\xi'_{\lambda,-} - \xi_{\lambda,-} = \mathbb{E}_{s\sim\pi', a\sim\mu'}[A^\mu_f(s,a)], \quad (10)$$

where $A^\mu_f(s,a)$ is the pseudo advantage with f as the reward function. With the aid of Equation 10, we can derive the performance difference formula of ξ_- as

$$\begin{aligned}\xi'_- - \xi_- &= (\xi'_{\lambda,-} - \xi_{\lambda,-}) + (\xi'_- - \xi'_{\lambda,-}) + (\xi_{\lambda,-} - \xi_-)\\ &= \mathbb{E}_{s\sim\pi', a\sim\mu'}[A^\mu_f(s,a)] - \beta\mathbb{E}_{s\sim\pi', a\sim\mu'}\left[(r-\eta')^2_- - (r-\lambda)^2_-\right]\\ &\quad - \beta\mathbb{E}_{s\sim\pi, a\sim\mu}\left[(r-\lambda)^2_- - (r-\eta)^2_-\right].\end{aligned}$$

Finally, by setting $\lambda = \eta$, we arrive at

$$\xi'_- - \xi_- = \mathbb{E}_{s\sim\pi', a\sim\mu'}[A^\mu_f(s,a)] - \beta\mathbb{E}_{s\sim\pi', a\sim\mu'}\left[(r-\eta')^2_- - (r-\eta)^2_-\right],$$

which is the same as Equation 8 if we explicitly calculate the advantage function with reward function f and $\lambda = \eta$. □

The MSVPDF in Equation 8 or Equation 11 claims that the MSV improvement can be separated into two parts. The first term in Equation 11 is a standard MDP with f as the reward function, and the second term is caused by the perturbation of the mean. It clearly quantifies the difficulty of solving the MSV problem, i.e., *the policy-dependent reward function breaks down the time-consistent nature of MDPs.* Meanwhile, it also shows us the standard MDP algorithm such as policy iteration (PI) is unavailable. A PI-like algorithm may be efficient in improving the first term, but the sign of the remaining term (dependent on η') is unpredictable. It suggests that we need novel tools to guarantee the policy improvement.

4.2 Performance Derivative Formula

While Equation 11 describes the performance difference between any two policies, we still need the local structure of the MSV problem to guide the direction of optimization. Following the line of the last part, we present the *MSV performance derivative formula* in this subsection, which describes the performance derivative at μ towards another policy μ'.

Theorem 4. *Given any two policies $\mu, \mu' \in \Pi$, we consider a mixed policy μ^ν,*

$$\mu^\nu(a \mid s) = (1-\nu)\mu(a \mid s) + \nu\mu'(a \mid s),$$

where the action follows μ with probability $1-\nu$, and follows μ' with probability ν for $\nu \in [0,1]$. We have

$$\frac{\mathrm{d}\xi_-}{\mathrm{d}\nu} = \mathbb{E}_{s\sim\pi, a\sim\mu'}[(1+2\beta\eta_-)A^\mu_\eta(s,a) - \beta A^\mu_{\zeta_-}(s,a)].$$

Proof. From MSVPDF, we obtain the difference for μ, μ^ν,

$$\xi^\nu_- - \xi_- = \mathbb{E}_{s\sim\pi^\nu, a\sim\mu^\nu}[A^\mu_f(s,a)] - \beta\mathbb{E}_{s\sim\pi^\nu, a\sim\mu^\nu}\left[(r-\eta^\nu)^2_- - (r-\eta)^2_-\right],$$

where $\eta^\nu := \eta^{\mu^\nu}$. Taking the derivative w.r.t. ν and letting $\nu \to 0$, we obtain the performance derivative formula. To simplify the derivation, we denote the terms on the right hand side as

$$\begin{aligned}
h_1(\nu) &= \mathbb{E}_{s\sim\pi^\nu, a\sim\mu^\nu}[A^\mu_f(s,a)],\\
h_2(\nu) &= \mathbb{E}_{s\sim\pi^\nu, a\sim\mu^\nu}\left[(r-\eta^\nu)^2_- - (r-\eta)^2_-\right].
\end{aligned}$$

Then $\xi^\nu_- - \xi_- = h_1(\nu) - \beta h_2(\nu)$. Specifically, we have

$$\begin{aligned}
h_1(\nu) &= \mathbb{E}_{s\sim\pi^\nu}[(1-\nu)\mathbb{E}_{a\sim\mu}[A^\mu_f(s,a)] + \nu\mathbb{E}_{a\sim\mu'}[A^\mu_f(s,a)]]\\
&= \nu\mathbb{E}_{s\sim\pi^\nu, a\sim\mu'}[A^\mu_f(s,a)],
\end{aligned}$$

where the last equality follows that $\mathbb{E}_{a\sim\mu}[A^\mu_f(s,a)] = 0$. Since $\lim_{\nu\to 0}\pi^\nu = \pi$, we obtain

$$\frac{\mathrm{d}h_1}{\mathrm{d}\nu} = \mathbb{E}_{s\sim\pi, a\sim\mu'}[A^\mu_f(s,a)].$$

Next, we differentiate $(r-\eta)^2_-$,

$$\begin{aligned}
\frac{\mathrm{d}(r-\eta)^2_-}{\mathrm{d}\nu} &= 2(r-\eta)_-\frac{\mathrm{d}(r-\eta^\nu)_-}{\mathrm{d}\nu}\\
&\overset{(i)}{=} -2(r-\eta)_-\mathbb{1}(r<\eta)\frac{\mathrm{d}\eta}{\mathrm{d}\nu} \qquad (11)\\
&\overset{(ii)}{=} -2(r-\eta)_-\frac{\mathrm{d}\eta}{\mathrm{d}\nu},
\end{aligned}$$

where (i) follows $\frac{\mathrm{d}(x)_-}{\mathrm{d}x} = \mathbb{1}(x < 0)$, and (ii) comes from $(r-\eta)_-\mathbb{1}(r<\eta) = (r-\eta)_-$. Thus, we have

$$\begin{aligned}
\frac{\mathrm{d}h_2}{\mathrm{d}\nu} &= \lim_{\nu\to 0}\frac{1}{\nu}\sum_s \pi^\nu(s)\sum_a \mu^\nu(a \mid s)\left[(r-\eta^\nu)_-^2 - (r-\eta)_-^2\right]\\
&= \lim_{\nu\to 0}\sum_s \pi^\nu(s)\sum_a \mu^\nu(a \mid s)\frac{(r-\eta^\nu)_-^2 - (r-\eta)_-^2}{\nu}\\
&= \sum_s \pi(s)\sum_a \mu(a \mid s)\frac{\mathrm{d}(r-\eta)_-^2}{\mathrm{d}\nu}\\
&= \sum_s \pi(s)\sum_a \mu(a \mid s)\left[-2(r-\eta)_-\frac{\mathrm{d}\eta}{\mathrm{d}\nu}\right]\\
&= -2\eta_-\frac{\mathrm{d}\eta}{\mathrm{d}\nu}.
\end{aligned}$$

Here we define the *semimean* η_- as

$$\eta_- := \eta_-^\mu = \mathbb{E}_{s\sim\pi, a\sim\mu}[(r-\eta)_-], \tag{12}$$

which is the downside expectation of rewards under π. From the standard result of PA (Cao, 2007, Chapter 2), we have

$$\frac{\mathrm{d}\eta}{\mathrm{d}\nu} = \mathbb{E}_{s\sim\pi, a\sim\mu'}[A_\eta^\mu(s,a)].$$

Putting the above relationships together, we obtain

$$\begin{aligned}
\frac{\mathrm{d}\xi_-}{\mathrm{d}\nu} &= \frac{\mathrm{d}h_1}{\mathrm{d}\nu} - \beta\frac{\mathrm{d}h_2}{\mathrm{d}\nu}\\
&= \mathbb{E}_{s\sim\pi, a\sim\mu'}[A_f^\mu(s,a)] + 2\beta\eta_-\frac{\mathrm{d}\eta}{\mathrm{d}\nu}\\
&= \mathbb{E}_{s\sim\pi, a\sim\mu'}[A_f^\mu(s,a) + 2\beta\eta_- A_\eta^\mu(s,a)]\\
&= \mathbb{E}_{s\sim\pi, a\sim\mu'}[(1+2\beta\eta_-)A_\eta^\mu(s,a) - \beta A_{\zeta_-}^\mu(s,a)].
\end{aligned}$$

□

The above equality indicates that the performance derivative is related to another reward function w.r.t. f (cf. Equation 9):

$$\begin{aligned}
g(s,a) &:= f(s,a) + 2\beta\eta_- r && (13)\\
&= (1+2\beta\eta_-)r - \beta(r-\eta)_-^2, && (14)
\end{aligned}$$

and the derivative formula can be written as

$$\frac{\mathrm{d}\xi_-}{\mathrm{d}\nu} = \mathbb{E}_{s\sim\pi, a\sim\mu'}[A_g^\mu(s,a)], \tag{15}$$

where $A_g^\mu(s,a)$ is the advantage function w.r.t. g.

With the performance derivative formula, we define the local optimum for MSV and present the necessary condition for MSV optimality.

Definition 1. *For a policy μ, $\exists \bar{\nu} \in (0,1)$ and we always have $\xi_-^{\mu} \geq \xi_-^{\nu}, \forall \nu \in (0, \bar{\nu})$, then we say μ is a local optimum in the mixed policy space.*

Theorem 5. *The optimal policy of MSV can be found in the deterministic policy space, and satisfies the necessary condition*

$$\mu^*(a \mid s) = \delta \left(a \in \operatorname*{argmax}_{b \in \mathcal{A}} A_g^*(s, b) \right),$$

which implies that $A_g^*(s,a) \leq 0, \forall s \in \mathcal{S}, a \in \mathcal{A}$. Here δ denotes the Dirac delta function.

Proof. The theorem is a direct result of the derivative formula. The (local) optimality implies that if μ is a local optimum, we always have $\frac{\mathrm{d}\xi_-}{\mathrm{d}\nu} \leq 0$ for any direction in the policy space. Assuming there is a contradiction, where for a state s there exists $\mu(a \mid s) = \delta(a = a')$ for any $a' \notin \operatorname{argmax}_b A_g^{\mu}(s, b)$, we can always find a better policy in the mixed policy space along the derivative direction. □

5. Optimization and Algorithms

In this section, we propose two approaches to optimize MSV with the parameterized policy. We firstly extend the policy gradient method to MSV with the pseudo reward function (cf. Equation 13) in Section 4. Following the same idea, we propose a trust region method to solve the MSV problem, and prove the the lower bound for its performance improvement. The two approaches together establish an iterative framework to solve the MSV problem.

5.1 MSV Policy Gradient Method

Policy gradient theorem is an essential foundation of modern deep RL algorithms, such as Actor-Critic methods. Here we consider the policy μ parameterized by $\theta \in \Theta$, which can be implemented with any differentiable function. We first give the MSV Policy Gradient (MSVPG) theory formally as follows.

Theorem 6. *For a policy μ parameterized by θ, we have*

$$\nabla_\theta \xi_- = \mathbb{E}_{s \sim \pi, a \sim \mu}[\nabla_\theta \log \mu(a \mid s) A_g^{\mu}(s,a)]. \tag{16}$$

The policy gradient for MSV can be easily proved by PA, which follows the same lines of derivative formula. For the readers from the DRL community, we also provide an alternative proof based on (Sutton & Barto, 2018) in the appendix.

Proof. Consider two policies μ, μ' parameterized by θ, θ' respectively. Their performance difference is given as

$$\xi_-' - \xi_- = \mathbb{E}_{s \sim \pi', a \sim \mu'}[A_f^{\mu}(s,a)] - \beta \mathbb{E}_{s \sim \pi', a \sim \mu'} \left[(r - \eta')_-^2 - (r - \eta)_-^2 \right].$$

Let denote $\Delta\theta = \theta' - \theta$. Similar to the derivation in Section 4.2, we denote the terms of above equation

$$h_1(\Delta\theta) = \mathbb{E}_{s \sim \pi', a \sim \mu'}[A_f^{\mu}(s,a)],$$
$$h_2(\Delta\theta) = \mathbb{E}_{s \sim \pi', a \sim \mu'} \left[(r - \eta')_-^2 - (r - \eta)_-^2 \right].$$

We take the limit of $\xi'_- - \xi_-$ by letting $\theta' \to \theta$.

$$\begin{aligned}
\nabla_\theta h_1 &= \lim_{\Delta\theta \to 0} \frac{1}{\Delta\theta} \sum_s \pi'(s) \sum_a \left[\mu'(a \mid s) A_f^\mu(s,a)\right] \\
&\overset{(i)}{=} \lim_{\Delta\theta \to 0} \sum_s \pi'(s) \sum_a \frac{\mu'(a \mid s) - \mu(a \mid s)}{\Delta\theta} A_f^\mu(s,a) \\
&= \sum_s \pi(s) \sum_a \nabla_\theta \mu(a \mid s) A_f^\mu(s,a) \\
&\overset{(ii)}{=} \mathbb{E}_{s\sim\pi, a\sim\mu}\left[\nabla_\theta \log \mu(a \mid s) A_f^\mu(s,a)\right],
\end{aligned}$$

where (i) follows $\mathbb{E}_{a\sim\mu}[A_f^\mu(s,a)] = 0$ and (ii) comes from $\nabla_\theta \log \mu(a \mid s) = \dfrac{\nabla_\theta \mu(a \mid s)}{\mu(a \mid s)}$.

Similar to the derivation in Equation 11, we have

$$\begin{aligned}
\nabla_\theta h_2 &= \lim_{\Delta\theta \to 0} \sum_s \pi'(s) \sum_a \mu'(a \mid s) \frac{(r-\eta')_-^2 - (r-\eta)_-^2}{\Delta\theta} \\
&= \sum_s \pi(s) \sum_a \mu(a \mid s) \lim_{\Delta\theta \to 0} \frac{(r-\eta')_-^2 - (r-\eta)_-^2}{\Delta\theta} \\
&= \sum_s \pi(s) \sum_a \mu(a \mid s) \nabla_\theta (r-\eta)_-^2 \\
&= \sum_s \pi(s) \sum_a \mu(a \mid s)[-2(r-\eta)_- \nabla_\theta \eta] \\
&= -2\eta_- \nabla_\theta \eta.
\end{aligned}$$

Since $\nabla_\theta \eta = \mathbb{E}_{s\sim\pi, a\sim\mu}\left[\nabla_\theta \log \mu(a \mid s) A_\eta^\mu(s,a)\right]$, we combine the results together and give the gradient of ξ_-

$$\begin{aligned}
\nabla_\theta \xi_- &= \nabla_\theta h_1 - \beta \nabla_\theta h_2 \\
&= \mathbb{E}_{s\sim\pi, a\sim\mu}\left[\nabla_\theta \log \mu(a \mid s) A_f^\mu(s,a)\right] + 2\beta\eta_- \nabla_\theta \eta \\
&= \mathbb{E}_{s\sim\pi, a\sim\mu}\left[\nabla_\theta \log \mu(a \mid s) A_f^\mu(s,a) + 2\beta\eta_- A_\eta^\mu(s,a)\right] \\
&= \mathbb{E}_{s\sim\pi, a\sim\mu}\left[\nabla_\theta \log \mu(a \mid s) A_g^\mu(s,a)\right].
\end{aligned}$$

□

Here we present an Actor-Critic algorithm based on MSVPG, which is named MSVAC (see Algorithm 2). In addition to the parameterized policy, we maintain another parameterized function V_ϕ as the value function. Then, the advantage function is estimated with the generalized advantage estimation (GAE) (Schulman et al., 2016). Typically, we have

$$\hat{A}_g(s_n, a_n) = \sum_{t=n}^{N-1} \lambda^{t-n} \left(g(s_t, a_t) - \hat{g} + V_\phi(s_t) - V_\phi(s_{t+1})\right), \tag{17}$$

Algorithm 2 MSVAC

```
Input: $\alpha, \beta, K, N$
Initialize the policy with $\theta$ and the value with $\phi$ randomly.
Set $\hat{\eta} = 0$, $\hat{\eta}_- = 0$, $\hat{\zeta}_- = 0$.
for $k = 1, 2, \cdots, K$ do
    Execute policy $\mu_\theta$ for $N$ times to collect $\{(s_n, a_n, r_n, s_{n+1})\}_{n=0}^{N-1}$.
    Update $\hat{\eta} \leftarrow (1-\alpha)\hat{\eta} + \alpha \frac{1}{N} \sum_{n=0}^{N-1} r_n$.
    Update $\hat{\eta}_- \leftarrow (1-\alpha)\hat{\eta}_- + \alpha \frac{1}{N} \sum_{n=0}^{N-1} (r_n - \hat{\eta})_-$.
    Update $\hat{\zeta}_- \leftarrow (1-\alpha)\hat{\zeta}_- + \alpha \frac{1}{N} \sum_{n=0}^{N-1} (r_n - \hat{\eta})_-^2$.
    Compute $g(s_n, a_n)$ with Equation 13 at all timesteps and $\hat{g}$.
    Compute $\hat{A}_g(s_n, a_n)$ with Equation 17 at all timesteps.
    Update the $\theta$ with Equation 16.
    Update the $\phi$ with Equation 18.
end for
```

where λ is the hyper-parameter to trade-off bias and variance, and $\hat{g} = (1 + 2\beta\hat{\eta}_-)\hat{\eta} - \beta\hat{\zeta}_-$ is the estimation of average surrogate reward function. With $\hat{V}_n = V_\phi(s_n) + \hat{A}_g(s_n, a_n)$ as the target value, we update the value function with

$$\mathcal{L}_V(\phi) := \frac{1}{2N} \sum_{n=0}^{N-1} (V_\phi(s_n) - \hat{V}_n)^2. \tag{18}$$

5.2 MSV Trust Region Method

While PG has a concise form, it often suffers from the difficulty of selecting step-sizes and the sensitivity to initial points in practice, especially when it works with neural networks. To address these drawbacks, trust region method (Schulman et al., 2015) is proposed to solve a surrogate problem in a local trust region and perform an approximate policy iteration.

5.2.1 Monotonic Improvement Guarantee

We extend the idea of trust region in the standard MDP into MSV, and propose the MSV Trust Region Policy Optimization (MSVTRPO) method. In MSVTRPO, we iteratively solve the problem as below

$$\begin{aligned} &\max_{\mu_\theta} \mathcal{L}_g^\mu(\mu_\theta) \\ &\text{s.t. } \mathbb{E}_{s\sim\pi} D_{\mathrm{TV}}(\mu_\theta(\cdot \mid s) \parallel \mu(\cdot \mid s)) \leq \epsilon_\mu, \end{aligned} \tag{19}$$

where

$$\mathcal{L}_g^\mu(\mu_\theta) := \mathbb{E}_{s\sim\pi, a\sim\mu_\theta} \left[A_g^\mu(s,a)\right].$$

Remark 1. *The trust region method updates the policy via the direction of maximum derivative (cf. the performance derivative formula in Equation 15), constrained in the proximity* *policy space with the TV-divergence. In contrast, the standard policy iteration scheme updates the policy via the same direction without constraint, which breaks the monotonic improvement for MSV.*

Next, we will show that MSVTRPO enjoys an analogous performance improvement bound. When the trust region is tight enough, i.e., $\epsilon_\mu \to 0$, the lower bound is dominated by the first order term.

To complete the proof, we need following lemma to bound the state-action distributions. For a policy μ, we denote the steady state-action distribution as $\rho(s, a) := \pi(s)\mu(s, a)$. Then we have:

Lemma 1. *For any two policies $\mu, \mu' \in \Pi$, the difference of their steady state-action distributions ρ, ρ' is bounded by*

$$\|\rho' - \rho\|_1 \leq 2\kappa'\epsilon_\mu.$$

Proof.

$$\begin{aligned}
\|\rho' - \rho\|_1 &= \sum_{s,a} |\pi'(s)\mu'(a \mid s) - \pi(s)\mu(a \mid s)| \\
&\leq \sum_{s,a} |\pi'(s)\mu'(a \mid s) - \pi(s)\mu'(a \mid s)| + \sum_{s,a} |\pi(s)\mu'(a \mid s) - \pi(s)\mu(a \mid s)| \\
&= \sum_{s} |\pi'(s) - \pi(s)| + \sum_{s} \pi(s) \sum_{a} |\mu'(a \mid s) - \mu(a \mid s)| \\
&\leq 2\left((\kappa' - 1)\epsilon_\mu + \epsilon_\mu\right) = 2\kappa'\epsilon_\mu,
\end{aligned}$$

where the last inequality follows that $\|\pi'(s) - \pi(s)\|_1 \leq 2(\kappa' - 1)\epsilon_\mu$ (see proposition 2 in appendix shown by Ma et al., 2021). □

Theorem 7. *Let μ' be the solution to the problem defined by Equation 19. We have*

$$\xi' - \xi \geq \mathcal{L}_g^\mu(\mu') - 2(\kappa' - 1)\epsilon_g\epsilon_\mu - 12\beta(\kappa')^2 R_{\max}^2 \epsilon_\mu^2,$$

where $\epsilon_g = \max_s |\mathbb{E}_{a\sim\mu'}[A_g^\mu(s, a)]|$ and κ' is Kemeny's constant under μ'.

Proof. Again, we start our analysis from MSVPDF. Based on Equation 11, we have

$$\begin{aligned}
\xi_-' - \xi_- &= \mathbb{E}_{s\sim\pi',a\sim\mu'}[A_f^\mu(s, a)] - \beta\mathbb{E}_{s\sim\pi',a\sim\mu'}\left[(r - \eta')_-^2 - (r - \eta)_-^2\right] \\
&= \mathbb{E}_{s\sim\pi',a\sim\mu'}[A_f^\mu(s, a) + 2\beta\eta_- A_\eta^\mu(s, a)] - \mathbb{E}_{s\sim\pi',a\sim\mu'}[2\beta\eta_- A_\eta^\mu(s, a)] \\
&\quad - \beta\mathbb{E}_{s\sim\pi',a\sim\mu'}\left[(r - \eta')_-^2 - (r - \eta)_-^2\right] \\
&= \mathbb{E}_{s\sim\pi',a\sim\mu'}[A_g^\mu(s, a)] - 2\beta\eta_-(\eta' - \eta) - \beta\mathbb{E}_{s\sim\pi',a\sim\mu'}\left[(r - \eta')_-^2 - (r - \eta)_-^2\right],
\end{aligned}$$

where the last equation follows the difference formula of average reward,

$$\eta' - \eta = \mathbb{E}_{s\sim\pi',a\sim\mu'}[A_\eta^\mu(s, a)]. \tag{20}$$

The result indicates that the difference can be separated into two parts: the improvement by optimizing the surrogate problem (the first term), and the discrepancy by the change of

η (the rest terms). The insight of our proof is to show that the first term dominates the difference and the rest terms can be ignored in a tight trust region.

The first term can be tackled with the standard trust region method. With the lower bound of average trust region method in Equation 7, we have

$$\mathbb{E}_{s\sim\pi',a\sim\mu'}[A_g^{\mu}(s,a)] - \mathcal{L}_g^{\mu}(\mu') \geq -2(\kappa'-1)\epsilon_g\epsilon_{\mu}. \tag{21}$$

Now, we need to bound the rest terms. We have

$$\begin{aligned}
&2\eta_-(\eta'-\eta) + \mathbb{E}_{s\sim\pi',a\sim\mu'}\left[(r-\eta')_-^2 - (r-\eta)_-^2\right] \\
&= \mathbb{E}_{s\sim\pi,a\sim\mu}[2(r-\eta)_-(\eta'-\eta)] + \mathbb{E}_{s\sim\pi',a\sim\mu'}\left[(r-\eta')_-^2 - (r-\eta)_-^2\right] \\
&= \mathbb{E}_{s\sim\pi',a\sim\mu'}\left[(r-\eta')_-^2 - (r-\eta)_-^2 + 2(r-\eta)_-(\eta'-\eta)\right] \\
&\quad - 2(\eta'-\eta)\big(\mathbb{E}_{s\sim\pi',a\sim\mu'}(r-\eta)_- - \mathbb{E}_{s\sim\pi,a\sim\mu}(r-\eta)_-\big).
\end{aligned}$$

Denote $h := (r'-\eta')_-^2 - (r'-\eta)_-^2 + 2(r'-\eta)_-(\eta'-\eta)$. Considering all potential cases for the relationship between η, η' and h, we have

- If $r \geq \max\{\eta, \eta'\}$, $h = 0$.
- If $r < \min\{\eta, \eta'\}$, $h = (r-\eta')^2 - (r-\eta)^2 + 2(r-\eta)(\eta'-\eta) = (\eta'-\eta)^2$.
- If $\eta \leq r < \eta'$, $h = (r-\eta')^2 \leq (\eta'-\eta)^2$.
- If $\eta' \leq r < \eta$, we denote $c_0 = r - \eta' \geq 0$ and $c_1 = \eta - r > 0$. We have $h = -(r-\eta)^2 + 2(r-\eta)(\eta'-\eta) = c_1^2 + 2c_0c_1 \leq (c_0+c_1)^2 = (\eta'-\eta)^2$.

Synthesizing the above results, we conclude $0 \leq h \leq (\eta'-\eta)^2$. Thus we have

$$\mathbb{E}_{s\sim\pi',a\sim\mu'}\left[(r-\eta')_-^2 - (r-\eta)_-^2 + 2(r-\eta)_-(\eta'-\eta)\right] \leq (\eta'-\eta)^2. \tag{22}$$

With Lemma 1, we obtain that

$$|\eta'-\eta| = |\rho' r - \rho r| \leq \|\rho'-\rho\|_1 R_{\max} \leq 2\kappa'\epsilon_{\mu}R_{\max},$$

where the first inequality follows the Hölder's inequality. Similarly, we have

$$\begin{aligned}
&|\mathbb{E}_{s\sim\pi',a\sim\mu'}(r-\eta)_- - \mathbb{E}_{s\sim\pi,a\sim\mu}(r-\eta)_-| \qquad (23)\\
&= |\rho'(r-\eta)_- - \rho(r-\eta)_-| \qquad (24)\\
&\leq \|\rho'-\rho\|_1 R_{\max} \qquad (25)\\
&\leq 2\kappa'\epsilon_{\mu}R_{\max} \qquad (26)
\end{aligned}$$

where Equation 25 comes from that $0 \leq (r-\eta)_- \leq 2R_{\max}$. Substituting the previous results into Equation 22 and combining with Equation 21, we arrive at

$$\begin{aligned}
\xi'_- - \xi_- \geq\ & \mathcal{L}_g^{\mu}(\mu') - 2(\kappa-1)\epsilon_g\epsilon_{\mu} - \beta|(\eta'-\eta)^2| \\
& - 2\beta|\eta'-\eta|\left|\mathbb{E}_{s\sim\pi',a\sim\mu'}(r-\eta)_- - \mathbb{E}_{s\sim\pi,a\sim\mu}(r-\eta)_-\right| \\
\geq\ & \mathcal{L}_g^{\mu}(\mu') - 2(\kappa'-1)\epsilon_g\epsilon_{\mu} - 12\beta(\kappa')^2R_{\max}^2\epsilon_{\mu}^2.
\end{aligned}$$

□

Algorithm 3 MSVPO

```
Input: $\alpha, \beta, K, N, M$
Initialize the policy with $\theta$ and the value with $\phi$ randomly.
Set $\hat{\eta} = 0$, $\hat{\eta}_- = 0$, $\hat{\zeta}_- = 0$.
for $k = 1, 2, \cdots, K$ do
    Execute policy $\mu_\theta$ for $N$ times to collect $\{(s_n, a_n, r_n, s_{n+1})\}_{n=0}^{N-1}$.
    $\hat{\eta} \leftarrow (1-\alpha)\hat{\eta} + \alpha \frac{1}{N} \sum_{n=0}^{N-1} r_n$.
    $\hat{\eta}_- \leftarrow (1-\alpha)\hat{\eta}_- + \alpha \frac{1}{N} \sum_{n=0}^{N-1} (r_n - \hat{\eta})_-$.
    $\hat{\zeta}_- \leftarrow (1-\alpha)\hat{\zeta}_- + \alpha \frac{1}{N} \sum_{n=0}^{N-1} (r_n - \hat{\eta})_-^2$.
    Compute $g(s_n, a_n)$ with Equation 13 at all timesteps and $\hat{g}$.
    Compute $\hat{A}_g(s_n, a_n)$ with Equation 17 at all timesteps.
    Update the $\theta$ with equation 27 for $M$ epochs.
    Update the $\phi$ with Equation 18 for $M$ epochs.
end for
```

5.2.2 Implementation details

In the end of this subsection, we address some implementation issues of MSVTRPO. First of all, in practice, we replace the TV-divergence with KL-divergence as most of trust region methods do. Since $D_{\mathrm{TV}}(p \parallel q) \leq \sqrt{D_{\mathrm{KL}}(p \parallel q)/2}$, the theoretical results are still applicable for the practical algorithms.

In the tabular case, where the state and action spaces are finite and discrete, it is enough to parameterize the policy tabularly. The previous analysis of TRPO (Abdolmaleki et al., 2018) shows that Equation 19 enjoys a closed form solution:

$$\mu'(\cdot \mid s, a) \propto \mu(\cdot \mid s, a) \exp\left(\frac{A_g^\mu(s,a)}{\upsilon^*}\right),$$

where υ^* can be obtained by solving the dual problem

$$\min_\upsilon \mathcal{L}(\upsilon) := \upsilon \epsilon_\mu + \upsilon \sum_s \pi(s) \log \sum_a \mu\left(a \mid s\right) \exp\left(\frac{A_g^\mu(s,a)}{\upsilon}\right).$$

With a known MDP, we name this iterative procedure as MSV Trust Region Policy Iteration (MSVTRPI). As aforementioned in Section 4, PI is not available for MSV. Nevertheless, we can do MSVTRPI as an alternative. When $\epsilon_\mu \to \infty$, it degrades to the standard PI without the monotonic improvement guarantee.

In the model-free case with large state and action spaces, we recommend solving the surrogate loss proposed by PPO (Schulman et al., 2017), for its stable performance and fast computing with neural networks. Formally, instead of optimizing the problem in Equation 19, we maximizing the clipping objective

$$\mathcal{L}_\mu^{\mathrm{CLIP}}(\theta) := \frac{1}{N} \sum_{n=0}^{N-1} \left[\min\Big(\omega_n(\theta)\hat{A}_g(s_n, a_n), \mathrm{clip}(\omega_n(\theta), 1-\varepsilon, 1+\varepsilon)\hat{A}_g(s_n, a_n)\Big)\right], \tag{27}$$

where $\omega_n(\theta) = \frac{\mu_\theta(a_n|s_n)}{\mu(a_n|s_n)}$ is the importance sampling ratio. Since we consider the long-run average performance in this paper, GAE is not applicable directly. Thus, we adopt the average value constraint (AVC) proposed by (Ma et al., 2021) to stabilize the value learning. The full algorithm, named by MSV Policy Optimization (MSVPO) is presented in Algorithm 3.

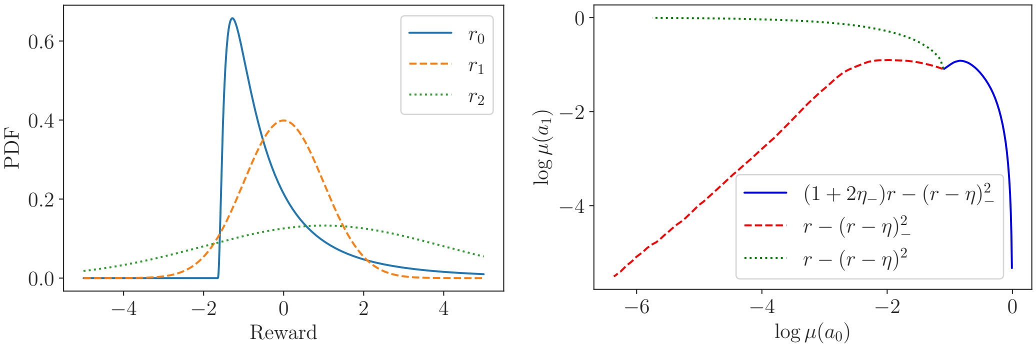
6. Experiments

In the previous sections, we analyze the properties of MSV problem and find that it can be soloved iteratively optimizing a surrogate reward function g (c.f. Equation 13). We also propose two methods to solve the MSV problem in the parameterized policy space.

To validate the effectiveness of our proposed methods in solving MSV problem, we conduct a series of experiments to answer the corresponding questions:

- Is the MSV really optimized by the the surrogate reward function g? Specifically, what is the difference from optimizing g instead of f?
- What is the difference between the MV (Xia, 2020) and MSV criteria?
- Does the proposed algorithms work well with the current deep RL algorithms?

6.1 Bandit Problem



(a) Reward distributions.

(b) Polices paths.

Figure 2: The bandit problem. (a) Reward distributions in the bandit problem. (b) Polices paths in the bandit problem. The paths are shown in the logarithmic parameter space.

We start from a simple bandit problem. In this problem, there are three actions with only a single state. Different actions result in different rewards following the distributions shown in Figure 2(a). Specifically, we have r_0 sampled from a shifted LogNormal$(0, 1)$ distribution, of which the mean is shifted to zero. If we choice a_1, we will obtain $r_1 \sim N(0, 2^2)$. Otherwise, we will have $r_2 \sim N(1, 3^2)$. Obviously, we have three different risk preference actions. When we fix $\beta = 1$ in MV and MSV, the agent should always choice a_0 if it optimizes the MSV criterion, and choice a_1 if it optimizes the MV criterion. The a_2 has the highest outcome, which is preferred by risk-neutral agents.

We compare three different agents, which optimize different reward functions. The first one optimizes $g = (1 + 2\eta_-)r - (r - \eta)_-^2$ (cf. Equation 13), which is the derived reward function with $\beta = 1$ in this work. The second one optimizes $f = r - (r - \eta)_-^2$ (cf. Equation 9),

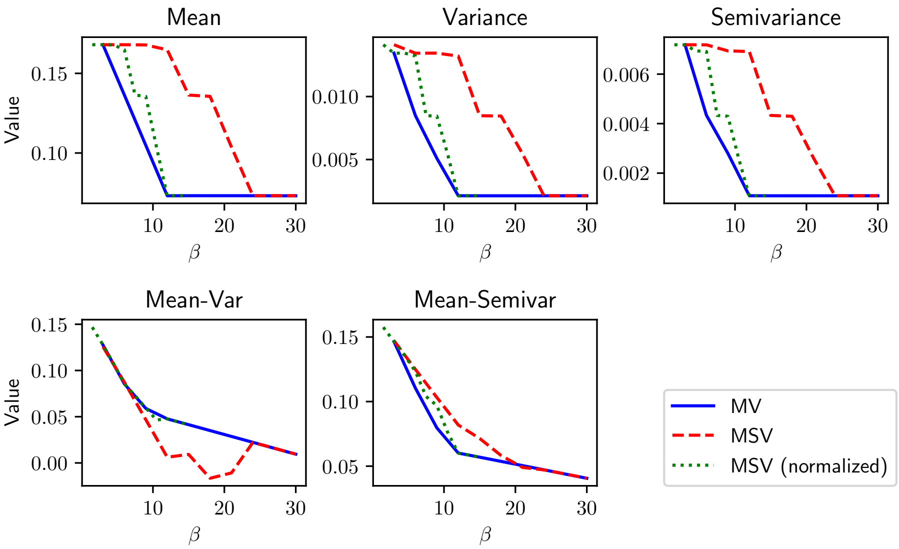


Figure 3: Comparison of MSVTRPI and MVPI in the portfolio management problem. The normalized MSV means β is doubled in comparison.

which is the Monte-Carlo return of MSV. We further consider a third agent which optimizes $r-(r-\eta)^2$ (Xia, 2020), an MV objective to illustrate the difference of MSV and MV problems. All the agents use policy gradient with a parameterized policy initialized as a uniform one.

To visualize the learning process, we plot the curves in the logarithmic parameter space, as shown in Figure 2(b). Since $\sum_i \mu(a_i) = 1$, $\mu(a_2)$ is ignored in the figure. As expected, the learning curve of the first agent (blue solid curve) approaches $(0, -\infty)$, meaning that it always chooses a_0 finally. Similarly, the third agent (green dotted curve) also chooses a_1 correspondingly. Interestingly, the second agent (red dashed curve), which optimizes the Monte-Carlo return of MSV, finally converges to choose a_2. The result tells us optimizing the reward $f = r - \beta(r-\eta)^2_-$ cannot optimize the MSV objective even in such a simple problem. This reflects the most essential difference between the optimization of policy-dependent reward and other problems. As discussed in Section 4, to optimize a problem with a policy-dependent reward function, we must consider the perturbation of the mean, at least in MSV problems.

6.2 Portfolio Management

In this part, we compare the performances of MSV- and MV-optimal polices in a portfolio management problem. We need to manage two independent assets and cash. At the stage t, the gain of the i-th asset is denoted by $x_{i,t} \in \{-0.2, -0.1, \ldots, 0.5\}$, which transits according to a transition probability matrix (described in Appendix). The action space is defined as $\mathcal{A} = \{(w_{1,t}, w_{2,t}) \mid \sum_{i=1,2} w_{i,t} \leq 1, w_{i,t} \in \{0, 0.2, \ldots, 1\}\}$, where $w_{i,t}$ is the weight of current portfolio on the i-th asset. Let $w_{0,t} = 1 - w_{1,t} - w_{2,t}$ denote the partition of cash in current portfolio and x_0 denote the return of cash. The reward function is defined as

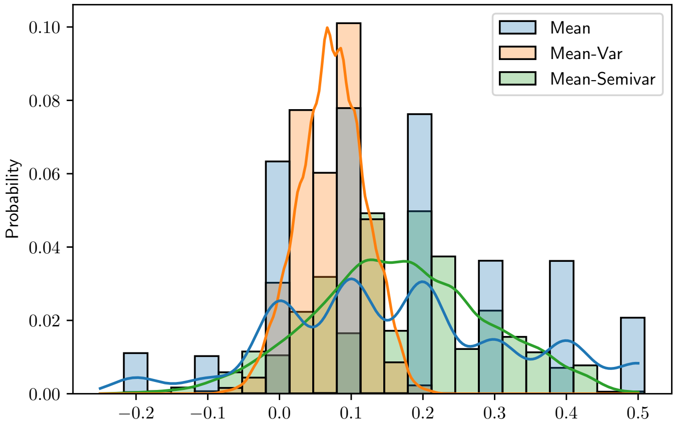


Figure 4: Reward distribution in the portfolio management problem. The policy optimizing MSV achieves $\eta = 0.168, \zeta = 0.014, \zeta_{-} = 0.006$. As a comparison, the policy optimizing MV achieves $\eta = 0.073, \zeta = 0.002, \zeta_{-} = 0.001$.

$r_t = w_{0,t}x_0 + \sum_{i=1,2} w_{i,t}x_{i,t} - \sum_{i=1,2} |w_{i,t} - w_{i,t-1}|c$, where c is the transition cost. The state is defined as $s_t = (x_{1,t}, x_{2,t}, w_{0,t}, w_{1,t})$. Hence, $|\mathcal{A}| = 21$ and $|\mathcal{S}| = 1344$.

For the MSV, we optimize the policy with the MSV trust region policy iteration (MSVTRPI) (see Section 5.2 for details), which aims to maximize $\xi_{-}^{\mu} = \eta^{\mu} - \beta\zeta_{-}^{\mu}$. We parameterize the policy in the softmax form as $\mu_\theta(a \mid s) := \text{softmax}(\theta(s,a))$, where $\theta \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$ are the "logic values". For MV, we optimize the policy with the mean-variance policy iteration (MVPI) proposed by Xia (2020), which maximizes $\xi^{\mu} = \eta^{\mu} - \beta\zeta^{\mu}$.

We change the risk preference parameter β and compare the MSVTRPI and MVPI. We depict the result in Figure 3, showing that with a fixed β, optimizing MSV always results in a larger return than that of MV. Besides, MV is more sensitive than MSV in terms of β, meaning that a small change of β will lead to a quick drop in both the return and risk. To better compare MSV and MV, we also show the "normalized" results of MSV, where we double β to provide the same penalty strength as MV. The result shows the normalized MSV also outperforms MV in terms of the average reward, illustrating that MSV is more plausible than MV. We demonstrate the reward distributions in Figure 4 with $\beta = 10$. It shows that MSV maintains high returns while avoiding large losses. In contrast, optimizing MV may be too conservative, as the upside rewards cause more volatility in this problem.

6.3 Robotic Control

To demonstrate the effectiveness of our proposed method in more general problem setups, we implement a "deep" variant algorithm named mean-semivariance policy optimization (MSVPO), which is based on the recent developed method APO (Ma et al., 2021) for average-reward RL problems.

We evaluate MSVPO in the continuous control benchmark MuJoCo (Todorov et al., 2012) with OpenAI gym (Brockman et al., 2016) as the interface. Since the original setup of MuJoCo is not suitable for the long-run average setting, we slightly modify the experimental protocol. In most of MuJoCo tasks, the agent will be terminated if it reaches any unsafe

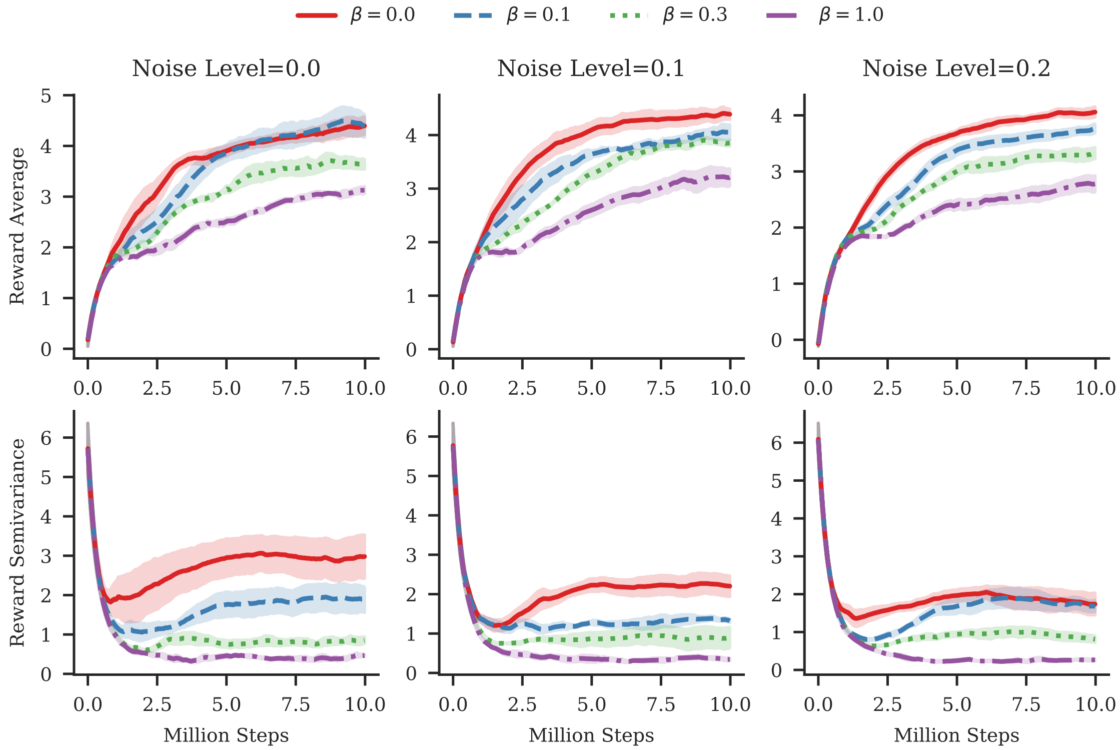


Figure 5: Training curves of Walker2d with noise. Each curve is averaged over 10 random seeds and shaded by the standard deviation.

state, such as falling down. In that cases, we will reset the system and add an extra cost to the terminal state. Different from other works focusing on the average episode returns, we are interested in the long-run average and semivariance of the steady reward distribution. To further increase the risk in the test scenarios, we add some noise to the agent outputs, i.e., the real action taken by the environment is $a_t + \epsilon$, where $\epsilon \sim N(0, \sigma^2)$. We call σ as the noise level of the modified MuJoCo tasks.

We evaluate MSVPO with different β's in the noisy Walker2d with different noise levels. When the agent falls, we penalize it with an extra cost -10 and reset the system. As shown in the Figure 5, the choice of different β's achieves the trade-off between the average and semivariance. In the noiseless environment (noise level $=$ 0), we interestingly find that risk-averse policy ($\beta = 0.1$) achieves competitive average reward with lower semivariance. It indicates that in complex scenes, optimizing a risk-averse metric may generate more robust policies with better performances comparing with a risk-neutral one.

To better understand the performance difference with different risk preference polices, we visualize the reward distributions of typical agents in Figure 6, where each agent of noise level 0.1 is evaluated for 1000 steps. We can see that risk-averse polices successfully avoid the unsafe states. Meanwhile, the agent uses smaller steps forward with the risk parameter β increasing. Instead, the risk-neutral agent tends to take the risk of falling for larger gains.

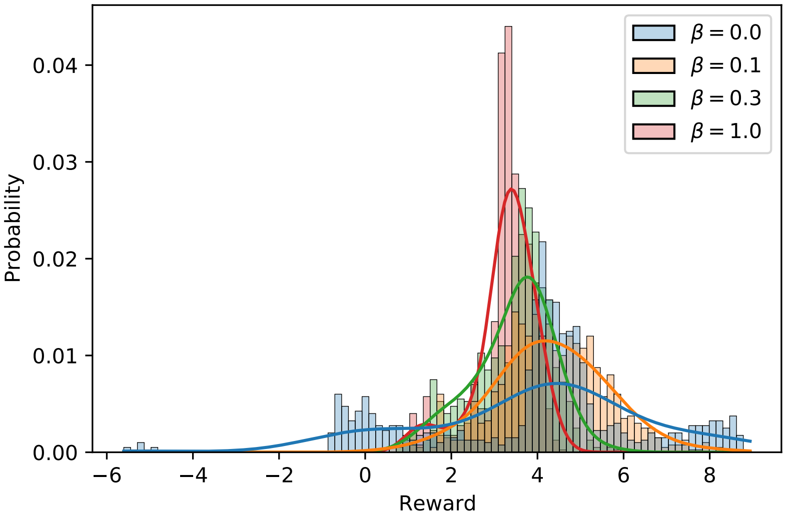


Figure 6: Reward distribution of Walker2d with noise.

7. Conclusion

This paper discusses how to optimize the mean-semivariance criterion for the steady reward of MDPs and RL, which is an alternative risk measure of mean-variance. The semivariance is a more reasonable measure than the variance in general scenarios, as it only penalizes the downside risk. We utilize PA theory to derive the performance difference formula and optimize MSV with data-driven approaches. We develop two algorithms for MSV based on PA theory, following the policy gradient theory and the trust region theory, respectively. We also demonstrate the effectiveness of the proposed algorithms in different problems, showing the risk-averse performance of MSV policy. We point out that the application of the proposed two-stage optimization framework for risk measures is not limited for MSV. We hope our work can promote the applications of data-driven approaches in risk-sensitive environments of MDPs and RL.

Acknowledgments

This work is funded by the National Natural Science Foundation of China (No. U1813216, 62192751, 61425027, 62073346, 11931018, U1811462), the National Key Research and Development Project of China under Grant 2017YFC0704100 and Grant 2016YFB0901900, in part by the 111 International Collaboration Program of China under Grant BP2018006, BNRist Program (BNR2019TD01009), the National Innovation Center of High Speed Train R&D project (CX/KJ-2020-0006), the Guangdong Province Key Laboratory of Computational Science at the Sun Yat-Sen University (2020B1212060032) and the Guangdong Basic and Applied Basic Research Foundation (2021A1515011984).

Appendix A. Brief Review of Perturbation Analysis theory

Consider an ergodic MDP with transition matrix P (induced by some policy μ), where $P(s' \mid s)$ is the transition probability from s to s'. We also consider a corresponding reward function r, where $r(s)$ is the reward expectation at s. We are interested in the average performance $\eta = \pi r$, where π denotes the steady state distribution. The Perturbation Analysis (PA) theory (Cao, 2007) captures how the performance changes if the policy (or system parameters P and r) has perturbations.

Theorem 8 (Performance difference formula)**.** *For two ergodic MDPs with P and P', we have*

$$\eta' - \eta = \pi[(P' - P)V + r' - r],$$

where V is the value function (called potential function in PA) for the system with P.

The value function satisfies the Poisson equation $(I - P)g + \eta e = r$, where I denotes the identity matrix and e is the unit vector.

Theorem 9 (Performance derivative formula)**.** *Consider another MDP with $P^\nu = P + \Delta P = (1-\nu)P + \nu P'$ and $r^\nu = r + \nu\Delta r = (1-\nu)r + \nu r'$. We have*

$$\left.\frac{\mathrm{d}\eta}{\mathrm{d}\nu}\right|_{\nu=0} = \pi[(\Delta P)V + \Delta r].$$

Appendix B. Alternative Proof of MSVPG

This proof follows the similar derivation of Sutton and Barto (2018, Chapter 13). We first derive the policy gradient of ζ_-, and give the complete form of MSV gradient by $\nabla_\theta \xi_- = \nabla_\theta \eta - \beta \nabla_\theta \zeta_-$. Taking the gradient of $V^\mu_{\zeta_-}$ for any arbitrary $s \in \mathcal{S}$, we have

$$\begin{aligned}
&\nabla_\theta V^\mu_{\zeta_-}(s) \\
&= \nabla_\theta \Big[\sum_a \mu(a \mid s) Q^\mu_{\zeta_-}(s,a) \Big] \\
&= \sum_a \Big[\nabla_\theta \mu(a \mid s) Q^\mu_{\zeta_-}(s,a) + \mu(a \mid s) \nabla_\theta Q^\mu_{\zeta_-}(s,a) \Big] \\
&= \sum_a \Big[\nabla_\theta \mu(a \mid s) Q^\mu_{\zeta_-}(s,a) + \mu(a \mid s) \nabla_\theta \sum_{s'} P\left(s' \mid s,a\right) \left((r-\eta)_-^2 - \zeta_- + V^\mu_{\zeta_-}\left(s'\right)\right) \Big] \\
&= \sum_a \Big[\nabla_\theta \mu(a \mid s) Q^\mu_{\zeta_-}(s,a) + \mu(a \mid s) \sum_{s'} P(s' \mid s,a)\big(-2(r-\eta)_- \nabla_\theta \eta - \nabla_\theta \zeta_- + \nabla_\theta V^\mu_{\zeta_-}\left(s'\right)\big) \Big].
\end{aligned}$$

Rephrasing the equation above, we obtain

$$\begin{aligned}
&\nabla_\theta \zeta_- = \\
&\sum_a \Big[\nabla_\theta \mu(a \mid s) Q^\mu_{\zeta_-}(s,a) + \mu(a \mid s) \sum_{s'} P\left(s' \mid s,a\right) \Big(\nabla_\theta V^\mu_{\zeta_-}\left(s'\right) - 2(r-\eta)_- \nabla_\theta \eta \Big) \Big] - \nabla_\theta V^\mu_{\zeta_-}(s).
\end{aligned}$$

Taking the expectation under π for both sides, we have

$$\begin{aligned}
&\nabla_\theta \zeta_- \\
&= \sum_s \pi(s) \sum_a \Big[\nabla_\theta \mu(a \mid s) Q^\mu_{\zeta_-}(s,a) + \mu(a \mid s) \sum_{s'} P\left(s' \mid s,a\right)\left(\nabla_\theta V^\mu_{\zeta_-}\left(s'\right) - 2(r-\eta)_- \nabla_\theta \eta\right)\Big] \\
&\quad - \sum_s \pi(s) \nabla_\theta V^\mu_{\zeta_-}(s) \\
&= \sum_s \pi(s) \sum_a \nabla_\theta \mu(a \mid s) Q^\mu_{\zeta_-}(s,a) + \sum_{s'} \sum_s \pi(s) \sum_a \mu(a \mid s) P\left(s' \mid s,a\right) \nabla_\theta V^\mu_{\zeta_-}\left(s'\right) \\
&\quad - \sum_s \pi(s) \sum_a \mu(a \mid s) \sum_{s'} 2(r-\eta)_- \nabla_\theta \eta - \sum_s \pi(s) \nabla_\theta V^\mu_{\zeta_-}(s). \qquad (28)
\end{aligned}$$

By the definitions of π and η_-, we have

$$\begin{aligned}
\pi\left(s'\right) &= \sum_s \pi(s) \sum_a \mu(a \mid s) P\left(s' \mid s,a\right), \\
\eta_- &= \sum_s \pi(s) \sum_a \mu(a \mid s) \sum_{s'} (r-\eta)_-.
\end{aligned}$$

Substituting into the Equation 28, we have

$$\begin{aligned}
\nabla_\theta \zeta_- &= \sum_s \pi(s) \sum_a \nabla_\theta \mu(a \mid s) Q^\mu_{\zeta_-}(s,a) + \sum_{s'} \pi\left(s'\right) \nabla_\theta V^\mu_{\zeta_-}\left(s'\right) - 2\eta_- \nabla_\theta \eta - \sum_s \pi(s) \nabla_\theta V^\mu_{\zeta_-}(s) \\
&= \sum_s \pi(s) \sum_a \nabla_\theta \mu(a \mid s) Q^\mu_{\zeta_-}(s,a) - 2\eta_- \nabla_\theta \eta \\
&= \sum_s \pi(s) \sum_a \nabla_\theta \mu(a \mid s) Q^\mu_{\zeta_-}(s,a) - 2\eta_- \sum_s \pi(s) \sum_a \nabla_\theta \mu(a \mid s) Q^\mu_\eta(s,a) \\
&= \sum_s \pi(s) \sum_a \nabla_\theta \mu(a \mid s) \Big[Q^\mu_{\zeta_-}(s,a) - 2\eta_- Q^\mu_\eta(s,a)\Big].
\end{aligned}$$

Finally, applying the trick $\nabla \log \mu = \nabla \mu / \mu$, we have

$$\nabla_\theta \zeta_- = \mathbb{E}_{s \sim \pi, a \sim \mu}\Big[Q^\mu_{\zeta_-}(s,a) - 2\eta_- Q^\mu_\eta(s,a)\Big].$$

Thus, the MSVPG is given by

$$\nabla_\theta \xi - = \mathbb{E}_{s \sim \pi, a \sim \mu}\Big[(1 + 2\eta_-) Q^\mu_\eta(s,a) - \beta Q^\mu_{\zeta_-}(s,a)\Big].$$

Appendix C. Experiment Details

C.1 The Setup of Portfolio Management Problem

The return of cash $x_0 = 0.01$. The transition cost $c = 0.05$.

Table 1: The transition matrix of asset 1

x_1	-0.2	-0.1	0	0.1	0.2	0.3	0.4	0.5
-0.2	0.09	0.05	0.25	0.24	0.18	0.05	0.10	0.04
-0.1	0.05	0.02	0.33	0.22	0.17	0.09	0.06	0.06
0	0.04	0.03	0.26	0.24	0.18	0.07	0.12	0.06
0.1	0.04	0.04	0.20	0.28	0.26	0.08	0.03	0.07
0.2	0.00	0.02	0.16	0.24	0.27	0.11	0.15	0.05
0.3	0.07	0.02	0.16	0.19	0.25	0.14	0.12	0.05
0.4	0.02	0.04	0.14	0.19	0.18	0.20	0.17	0.06
0.5	0.03	0.03	0.09	0.19	0.23	0.15	0.14	0.14

Table 2: The transition matrix of asset 2

x_2	-0.2	-0.1	0	0.1	0.2	0.3	0.4	0.5
-0.2	0.13	0.10	0.08	0.09	0.20	0.36	0.02	0.02
-0.1	0.06	0.11	0.09	0.12	0.17	0.37	0.04	0.04
0	0.01	0.06	0.12	0.15	0.25	0.35	0.02	0.04
0.1	0.06	0.06	0.12	0.15	0.22	0.34	0.01	0.04
0.2	0.02	0.04	0.09	0.24	0.23	0.32	0.04	0.02
0.3	0.04	0.07	0.11	0.20	0.26	0.27	0.03	0.02
0.4	0.10	0.11	0.13	0.16	0.17	0.20	0.04	0.09
0.5	0.01	0.10	0.30	0.21	0.16	0.16	0.00	0.06

Appendix D. Hyper-parameters of MSVPO

Hyper-parameter	Value
Network learning rate β	3e-4
Network hidden sizes	[64, 64]
Activation function	Tanh
Optimizer	Adam
Batch size	256
Gradient Clipping	10
Clipping parameter ε	0.2
Optimization Epochs M	10
GAE parameter λ	0.95
Average Value Constraint Coefficient in APO (Ma et al., 2021) ν	0.3

Table 3: Hyper-parameters sheet

References

Abdolmaleki, A., Springenberg, J. T., Tassa, Y., Munos, R., Heess, N., & Riedmiller, M. A. (2018). Maximum a posteriori policy optimisation. In *International Conference on Learning Representations*.

Achiam, J., Held, D., Tamar, A., & Abbeel, P. (2017). Constrained policy optimization. In *International Conference on Machine Learning*, Vol. 70, pp. 22–31.

Berner, C., Brockman, G., Chan, B., Cheung, V., Dębiak, P., Dennison, C., Farhi, D., Fischer, Q., Hashme, S., Hesse, C., et al. (2019). Dota 2 with large scale deep reinforcement learning. *ArXiv preprint*, *abs/1912.06680*.

Bisi, L., Sabbioni, L., Vittori, E., Papini, M., & Restelli, M. (2020). Risk-averse trust region optimization for reward-volatility reduction. In *International Joint Conference on Artificial Intelligence*, pp. 4583–4589.

Bollerslev, T., Li, S. Z., & Zhao, B. (2020). Good volatility, bad volatility, and the cross section of stock returns. *Journal of Financial and Quantitative Analysis*, *55*(3), 751–781.

Borkar, V. S., & Meyn, S. P. (2002). Risk-sensitive optimal control for Markov decision processes with monotone cost. *Mathematics of Operations Research*, *27*(1), 192–209.

Briec, W., & Kerstens, K. (2009). Multi-horizon markowitz portfolio performance appraisals: A general approach. *Omega*, *37*(1), 50–62.

Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., & Zaremba, W. (2016). OpenAI Gym. *ArXiv preprint*, *abs/1606.01540*.

Cao, X.-R. (2007). *Stochastic Learning and Optimization: A Sensitivity-Based Approach*. Springer.

Castro, D. D., Tamar, A., & Mannor, S. (2012). Policy gradients with variance related risk criteria. In *International Conference on Machine Learning*, pp. 1651–1658.

Chen, W., Li, D., Lu, S., & Liu, W. (2019). Multi-period mean–semivariance portfolio optimization based on uncertain measure. *Soft Computing*, *23*(15), 6231–6247.

Choobineh, F., & Branting, D. (1986). A simple approximation for semivariance. *European Journal of Operational Research*, *27*(3), 364–370.

Chow, Y., & Ghavamzadeh, M. (2014). Algorithms for CVaR optimization in MDPs. In *Advances in Neural Information Processing Systems*, pp. 3509–3517.

Chow, Y., Ghavamzadeh, M., Janson, L., & Pavone, M. (2017). Risk-constrained reinforcement learning with percentile risk criteria. *Journal of Machine Learning Research*, *18*(1), 6070–6120.

Chow, Y., Tamar, A., Mannor, S., & Pavone, M. (2015). Risk-sensitive and robust decision-making: a CVaR optimization approach. In *Advances in Neural Information Processing Systems*, pp. 1522–1530.

Chung, K.-J. (1994). Mean-variance tradeoffs in an undiscounted MDP: the unichain case. *Operations Research*, *42*(1), 184–188.

Delage, E., Kuhn, D., & Wiesemann, W. (2019). "Dice"-sion–making under uncertainty: When can a random decision reduce risk?. *Management Science*, *65*(7), 3282–3301.

Dulac-Arnold, G., Mankowitz, D., & Hester, T. (2019). Challenges of real-world reinforcement learning. *ArXiv preprint*, *abs/1904.12901*.

Estrada, J. (2007). Mean-semivariance behavior: Downside risk and capital asset pricing. *International Review of Economics & Finance*, *16*(2), 169–185.

Fei, Y., Yang, Z., Chen, Y., Wang, Z., & Xie, Q. (2020). Risk-sensitive reinforcement learning: Near-optimal risk-sample tradeoff in regret. In *Advances in Neural Information Processing Systems*, Vol. 33, pp. 22384–22395.

Filar, J. A., Kallenberg, L. C., & Lee, H.-M. (1989). Variance-penalized Markov decision processes. *Mathematics of Operations Research*, *14*(1), 147–161.

Garcıa, J., & Fernández, F. (2015). A comprehensive survey on safe reinforcement learning. *Journal of Machine Learning Research*, *16*(1), 1437–1480.

Gosavi, A. (2014). Variance-penalized Markov decision processes: Dynamic programming and reinforcement learning techniques. *International Journal of General Systems*, *43*(6), 649–669.

Hogan, W. W., & Warren, J. M. (1974). Toward the development of an equilibrium capital-market model based on semivariance. *Journal of Financial and Quantitative Analysis*, *9*(1), 1–11.

Howard, R. A., & Matheson, J. E. (1972). Risk-sensitive Markov decision processes. *Management science*, *18*(7), 356–369.

Janner, M., Fu, J., Zhang, M., & Levine, S. (2019). When to trust your model: Model-based policy optimization. In *Advances in Neural Information Processing Systems*, pp. 12498–12509.

Li, D., & Ng, W.-L. (2000). Optimal dynamic portfolio selection: Multiperiod mean-variance formulation. *Mathematical Finance*, *10*(3), 387–406.

Liu, Y.-J., & Zhang, W.-G. (2015). A multi-period fuzzy portfolio optimization model with minimum transaction lots. *European Journal of Operational Research*, *242*(3), 933–941.

Ma, S., Ma, X., & Xia, L. (2022a). An optimistic value iteration for mean–variance optimization in discounted markov decision processes. *Results in Control and Optimization*, *8*, 100165.

Ma, S., Ma, X., & Xia, L. (2022b). A unified algorithm framework for mean-variance optimization in discounted Markov decision processes. *ArXiv preprint*, *abs/2201.05737*.

Ma, X., Tang, X., Xia, L., Yang, J., & Zhao, Q. (2021). Average-reward reinforcement learning with trust region methods. In *International Joint Conference on Artificial Intelligence*, pp. 2797–2803.

Ma, X., Xia, L., Zhou, Z., Yang, J., & Zhao, Q. (2020). Dsac: distributional soft actor critic for risk-sensitive reinforcement learning. *ArXiv preprint*, *abs/2004.14547*.

Markowitz, H., Todd, P., Xu, G., & Yamane, Y. (1993). Computation of mean-semivariance efficient sets by the critical line algorithm. *Annals of Operations Research*, *45*(1), 307–317.

Markowitz, H. M. (1952). Portfolio selection. *Journal of Finance*, *7*, 77–91.

Markowitz, H. M. (1959). *Portfolio Selection: Efficient Diversification of Investments*. John Wiley & Sons, New York.

Mavrin, B., Yao, H., Kong, L., Wu, K., & Yu, Y. (2019). Distributional reinforcement learning for efficient exploration. In *International Conference on Machine Learning*, Vol. 97, pp. 4424–4434.

Nagabandi, A., Konolige, K., Levine, S., & Kumar, V. (2020). Deep dynamics models for learning dexterous manipulation. In *Conference on Robot Learning*, pp. 1101–1112.

Nemirovski, A., & Shapiro, A. (2007). Convex approximations of chance constrained programs. *SIAM Journal on Optimization*, *17*(4), 969–996.

Prashanth, L., & Ghavamzadeh, M. (2016). Variance-constrained actor-critic algorithms for discounted and average reward MDPs. *Machine Learning*, *105*(3), 367–417.

Ruszczyński, A. (2010). Risk-averse dynamic programming for Markov decision processes. *Mathematical programming*, *125*(2), 235–261.

Schulman, J., Levine, S., Abbeel, P., Jordan, M. I., & Moritz, P. (2015). Trust region policy optimization. In *International Conference on Machine Learning*, Vol. 37, pp. 1889–1897.

Schulman, J., Moritz, P., Levine, S., Jordan, M. I., & Abbeel, P. (2016). High-dimensional continuous control using generalized advantage estimation. In *International Conference on Learning Representations*.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal policy optimization algorithms. *ArXiv preprint*, *abs/1707.06347*.

Shapiro, A., Dentcheva, D., & Ruszczynski, A. (2021). *Lectures on stochastic programming: modeling and theory*. SIAM.

Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, A., et al. (2017). Mastering the game of go without human knowledge. *Nature*, *550*(7676), 354–359.

Sobel, M. J. (1982). The variance of discounted Markov decision processes. *Journal of Applied Probability*, *19*(4), 794–802.

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement Learning: An Introduction.* MIT press.

Tamar, A., Chow, Y., Ghavamzadeh, M., & Mannor, S. (2016). Sequential decision making with coherent risk. *IEEE Transactions on Automatic Control*, *62*(7), 3323–3338.

Tamar, A., Glassner, Y., & Mannor, S. (2015). Optimizing the CVaR via sampling. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, pp. 2993–2999.

Todorov, E., Erez, T., & Tassa, Y. (2012). Mujoco: A physics engine for model-based control. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5026–5033.

Vinyals, O., Babuschkin, I., Czarnecki, W. M., Mathieu, M., Dudzik, A., Chung, J., Choi, D. H., Powell, R., Ewalds, T., Georgiev, P., et al. (2019). Grandmaster level in starcraft ii using multi-agent reinforcement learning. *Nature*, *575*(7782), 350–354.

Wei, Q. (2019). Mean–semivariance optimality for continuous-time Markov decision processes. *Systems & Control Letters*, *125*, 67–74.

Xia, L. (2016). Optimization of Markov decision processes under the variance criterion. *Automatica*, *73*, 269–278.

Xia, L. (2020). Risk-sensitive Markov decision processes with combined metrics of mean and variance. *Production and Operations Management*, *29*(12), 2808–2827.

Xie, T., Liu, B., Xu, Y., Ghavamzadeh, M., Chow, Y., Lyu, D., & Yoon, D. (2018). A block coordinate ascent algorithm for mean-variance optimization. In *Advances in Neural Information Processing Systems*, Vol. 31, pp. 1073–1083.

Yan, W., Miao, R., & Li, S. (2007). Multi-period semi-variance portfolio selection: Model and numerical solution. *Applied Mathematics and Computation*, *194*(1), 128–134.

Zhang, S., Liu, B., & Whiteson, S. (2021). Mean-variance policy iteration for risk-averse reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35, pp. 10905–10913.

Zhang, W.-G., Liu, Y.-J., & Xu, W.-J. (2012). A possibilistic mean-semivariance-entropy model for multi-period portfolio selection with transaction costs. *European Journal of Operational Research*, *222*(2), 341–349.

Zhang, Y., & Ross, K. W. (2021). On-policy deep reinforcement learning for the average-reward criterion. In *International Conference on Machine Learning*, Vol. 139, pp. 12535–12545.

Zhou, F., Wang, J., & Feng, X. (2020). Non-crossing quantile regression for distributional reinforcement learning. In *Advances in Neural Information Processing Systems*, Vol. 33, pp. 15909–15919.