Continuous-Time Reinforcement Learning for Asset–Liability Management

Yilie Huang
yh2971@columbia.edu
Department of Industrial Engineering and Operations Research, Columbia University
New York, New York, USA

Abstract

This paper proposes a novel approach for Asset–Liability Management (ALM) by employing continuous-time Reinforcement Learning (RL) with a linear-quadratic (LQ) formulation that incorporates both interim and terminal objectives. We develop a model-free, policy gradient-based soft actor-critic algorithm tailored to ALM for dynamically synchronizing assets and liabilities. To ensure an effective balance between exploration and exploitation with minimal tuning, we introduce adaptive exploration for the actor and scheduled exploration for the critic. Our empirical study evaluates this approach against two enhanced traditional financial strategies, a model-based continuous-time RL method, and three state-of-the-art RL algorithms. Evaluated across 200 randomized market scenarios, our method achieves higher average rewards than all alternative strategies, with rapid initial gains and sustained superior performance. The outperformance stems not from complex neural networks or improved parameter estimation, but from directly learning the optimal ALM strategy without learning the environment.

Keywords

Continuous-time Reinforcement Learning, Asset-Liability Management, Model-Free, Actor-Critic

1 Introduction

Asset-Liability Management (ALM) [36] is a critical component of financial strategy, involving the careful coordination of assets and liabilities to ensure the financial health of institutions. It plays a crucial role for banks, insurance companies, and pension funds, where the alignment of assets against liabilities significantly affects financial stability and regulatory compliance.

Traditionally, ALM has utilized a range of methods to effectively synchronize assets and liabilities: static approaches like cash flow matching [39] ensure liabilities are met with corresponding asset inflows, and passive value-driven strategies such as key rate duration matching [10] mitigate interest rate fluctuations by aligning the durations of assets and liabilities. On the other hand, more dynamic techniques, such as contingent immunization [18, 19] and Constant Proportion Portfolio Insurance (CPPI) [4, 9], actively adjust asset allocation to maintain a target surplus or minimize deviation from a predefined target. Maintaining a target surplus is essential for balancing solvency and efficient capital use, helping to mitigate both the risk of insolvency from insufficient surplus and the inefficiency of excess capital. However, these traditional methods often assume a stable environment with complete information, which limits their adaptability in fast-changing market conditions.

Reinforcement learning (RL) offers notable advantages in ALM by dynamically adjusting policies based on real-time feedback, making it a powerful tool for decision-making in dynamic and uncertain environments. Despite its potential, most RL methods have been developed around discrete-time Markov decision processes (MDPs) with discrete state and action spaces. While effective in many domains, this discrete-time framework faces inherent limitations when applied to systems that naturally evolve in continuous time, such as financial markets. Bridging this gap requires discretizing continuous-time problems into discrete-time models, enabling the use of standard RL algorithms. However, this discretization introduces critical challenges. Selecting an appropriate time step size is particularly challenging: large time steps can oversimplify the problem, reducing resolution and yielding suboptimal policies, while small time steps, despite their precision, increase the computational burden and may result in instability [23, 25, 33]. Additionally, discretization often struggles to accurately capture complex, fine-grained dynamics in continuous environments, especially in high-frequency decision-making contexts. This mismatch between discrete-time models and continuous-time dynamics limits the effectiveness of traditional RL methods in such settings, underscoring the need for continuous-time RL frameworks that can better handle the inherent complexities of financial markets.

Recent advancements in continuous-time RL have marked a pivotal shift from earlier isolated studies [2, 7, 35] to a more cohesive and systematic framework. The introduction of an entropy-regularized control approach by [37] laid a rigorous mathematical foundation for this field. Building on this, subsequent research [16, 17] developed comprehensive methods for policy evaluation and improvement. A key aspect of this research is its model-free, data-driven approach, which focuses on directly learning optimal control policies without requiring explicit model estimation. This progress has not only solidified the theoretical underpinnings of continuous-time RL but also inspired diverse extensions and practical applications in domains requiring robust real-time decision-making under uncertainty [12, 13, 34, 38].

While recent developments in continuous-time RL have made significant theoretical progress, much of the existing work remains primarily focused on analytical results rather than empirical validation. To date, these methods have not been applied to ALM, a setting that naturally fits the continuous-time framework due to its dynamic and stochastic nature. Furthermore, empirical comparisons between continuous-time RL and traditional discrete-time RL approaches in financial applications remain largely unexplored, leaving open the question of their relative practical effectiveness.

Building upon these theoretical advancements, our work mainly contributes in the following ways:

- A linear-quadratic (LQ) formulation of the ALM problem is introduced, incorporating both interim and terminal objectives, and addressing the time-inconsistency and terminal constraint issues in the traditional mean-variance (MV) approach.
- We design a model-free, policy gradient-based (soft) actor–critic algorithm tailored for the ALM problem within a continuous-time RL framework. To the best of our knowledge, this is the first work that applies continuous-time RL to ALM.
- We introduce an adaptive exploration mechanism for the actor and a scheduled exploration strategy for the critic, enabling a robust exploration–exploitation trade-off throughout learning.
- Empirical results demonstrate that our algorithm achieves superior performance over traditional ALM strategies, model-based continuous-time RL, and advanced RL baselines.

The remainder of the paper is organized as follows. Section 2 formulates the ALM problem and discusses preliminary results essential for subsequent developments. Section 3 outlines our RL algorithm's development and principles. Section 4 proves the convergence of the proposed algorithm. Section 5 evaluates our algorithm against six ALM strategies and presents experimental results. Finally, Section 6 concludes.

2 Description of the Asset-Liability Management Problem

2.1 Classical Stochastic LQ Framework for ALM

One widely used approach to address the ALM problem is through stochastic control, particularly using the MV formulation, as demonstrated in several studies [11, 24, 30]. In this framework, the state variable is typically defined as the surplus, representing the difference between assets and liabilities. The primary objective is to minimize the expected squared deviation of the terminal surplus from a predefined target surplus, subject to a terminal constraint.

While the MV formulation provides a well-structured framework for ALM, it also presents several key limitations. First, MV-based formulations suffer from an inherent time-inconsistency issue, meaning that strategies deemed optimal at the outset may become suboptimal as time progresses [43]. Second, addressing the terminal constraint in MV formulations often requires additional techniques, such as introducing Lagrange multipliers, which can complicate the solution. Finally, by focusing exclusively on minimizing the terminal surplus deviation, the MV approach overlooks the importance of managing the surplus throughout the entire horizon, which is crucial for ensuring financial stability and meeting ongoing obligations.

To address these limitations, we formulate the ALM problem as a specific stochastic LQ control problem, which is structurally similar to the MV approach but introduces key differences. The first distinction lies in the state representation: instead of directly modeling the surplus, we define the state variable $x(t) \in \mathbb{R}$ as the surplus deviation, representing the difference between the surplus and the target surplus:

$$x(t) = \text{Assets}(t) - \text{Liabilities}(t) - \text{Target Surplus},$$

where a positive $x(t)$ indicates a surplus above the target, implying inefficient capital use, while a negative $x(t)$ represents a shortfall, increasing the risk of insolvency.

By modeling the deviation directly, this formulation simplifies the ALM problem and provides a clearer objective aligned with maintaining a stable surplus relative to a predefined target. It also allows for more precise tracking of the financial position by explicitly focusing on deviations rather than the raw surplus deviation.

The control variable $u(t) \in \mathbb{R}$ represents strategic financial decisions, such as asset reallocation, funding adjustments, and liability management, aimed at minimizing deviations from the target surplus over time.

The dynamics of surplus deviation x under the influence of financial control u follow the stochastic differential equation (SDE):

$$\mathrm{d}x^u(t) = (Ax^u(t) + Bu(t))\mathrm{d}t + (Cx^u(t) + Du(t))\mathrm{d}W(t), \quad (1)$$

where $x^u(0) = x_0$ is the initial surplus deviation and $W(t)$ is standard Brownian motion, representing the stochastic nature of financial markets. The model parameters are interpreted as follows:

- A: Represents the internal drift, modeling the natural tendency of the surplus deviation to increase or decrease over time without intervention.
- B: Captures the direct impact of the control u on the surplus deviation. A larger B means control actions have a stronger influence.
- C: Scales the impact of the current surplus deviation on the volatility of the system. Higher values of C amplify how fluctuations in the surplus deviation contribute to uncertainty in the dynamics.
- D: Describes how control actions affect the variability of the surplus deviation. Higher values of D imply that control actions have a stronger impact on the uncertainty in the system.

Our objective is to manage u strategically to minimize deviations from the target surplus, penalizing both positive and negative deviations. Positive deviations imply surplus accumulation beyond the target, which could result in inefficient capital use or reduced financial efficiency. Negative deviations, on the other hand, indicate a shortfall relative to the target, reducing the safety buffer against financial uncertainties. We aim to optimize the expected value of the quadratic objective functional that incorporates both interim and terminal deviations:

$$\max_u \mathbb{E}\left[\int_0^T -\frac{1}{2}Qx^u(t)^2\mathrm{d}t - \frac{1}{2}Hx^u(T)^2\right], \quad (2)$$

where $Q \geq 0$ and $H \geq 0$ are coefficients that penalize deviations from the target surplus over the time horizon $[0, T]$ and at the terminal time T, respectively.

Notably, unlike conventional formulations, we do not explicitly penalize the control u in the objective function. Instead, u is implicitly constrained through the dynamics given in (1). This type of formulation has led to an active research area known as "indefinite stochastic Linear Quadratic control" [5, 26].

Provided the model parameters A, B, C, D, Q, and H are known, the established stochastic control theory can solve this optimization problem [41], producing an optimal value function and control

policy:

$$V^{CL}(t,x) = -\frac{1}{2}\left[\frac{Q}{\Lambda} + (H - \frac{Q}{\Lambda})e^{\Lambda(t-T)}\right]x^2,$$
$$u^{CL}(t,x) = -\frac{B+CD}{D^2}x, \tag{3}$$

where $\Lambda = \frac{1}{D^2}(B^2 + 2BCD - 2AD^2)$.

2.2 Continuous-Time RL Framework for ALM

However, the complete and precise knowledge of parameters such as A, B, C, and D in real-life ALM scenarios is often impractical, necessitating the use of RL to manage uncertainties. RL addresses these challenges by maintaining a balance between exploration and exploitation, adapting dynamically to the unknown parameters of the environment [31]. This is achieved through randomized control processes, where controls u are derived from a distribution $\pi = \{\pi(\cdot,t) \in \mathcal{P}(\mathbb{R}) : 0 \le t \le T\}$, representing all probability density functions over $\mathbb{R}$. To encourage exploration, an entropy term is integrated into the objective function, promoting stochastic policies. This approach is conceptually related to soft-max approximations and Boltzmann exploration strategies [8, 44].

By [37], under entropy-regularized RL for continuous-time controlled diffusion processes, the dynamics of the ALM problem under stochastic policy π are given by:

$$\mathrm{d}x^{\pi}(t) = \widetilde{b}(x^{\pi}(t), \pi(\cdot,t))\mathrm{d}t + \widetilde{\sigma}(x^{\pi}(t), \pi(\cdot,t))\mathrm{d}W(t), \tag{4}$$

where the drift $\widetilde{b}$ and diffusion $\widetilde{\sigma}$ components are defined as:

$$\widetilde{b}(x,\pi) = Ax + B\int_{\mathbb{R}} u\pi(u)\mathrm{d}u,$$
$$\widetilde{\sigma}(x,\pi) = \sqrt{\int_{\mathbb{R}}(Cx+Du)^2\pi(u)\mathrm{d}u}, \quad (x,\pi) \in \mathbb{R}\times\mathcal{P}(\mathbb{R}). \tag{5}$$

The entropy-regularized value function for stochastic policy π is expressed as:

$$J(t,x;\pi) = \mathbb{E}\left[\int_t^T \left(-\frac{1}{2}Qx^{\pi}(s)^2 + \gamma p(s)\right)\mathrm{d}s - \frac{1}{2}Hx^{\pi}(T)^2 \middle| x^{\pi}(t) = x\right], \tag{6}$$

where $p(t) = -\int_{\mathbb{R}} \pi(t,u)\log\pi(t,u)\mathrm{d}u$ represents the entropy term, and γ, known as the temperature parameter, is the weight on exploration.

The optimal value function and optimal randomized/stochastic (feedback) policy are solved as follows:

$$V(t,x) = -\frac{1}{2}k_1(t)x^2 + k_3(t),$$
$$\pi(u \mid t,x) = \mathcal{N}\left(u \middle| -\frac{(B+CD)}{D^2}x, \frac{\gamma}{D^2k_1(t)}\right), \tag{7}$$

where $k_1 > 0$ and k_3 are certain functions of t that can be determined completely by the model parameters.

It should be noted that the values of all model parameters are unknown to the agent, meaning that the optimal solutions in (7) cannot be directly applied. Moreover, we make no attempt to estimate these parameters, as is typically done in model-based approaches. Instead, we adopt a model-free approach that entirely avoids model estimation. Despite the unknown parameters, this model provides critical insights into the structural properties of the optimal solutions, thereby reducing the complexity of function parameterization and approximation in the learning process. This advantage will be demonstrated in the next section.

3 A Continuous-Time RL Algorithm

This section presents a continuous-time RL algorithm specifically designed for ALM. It covers critical aspects including function parameterization, policy evaluation and improvement methods, adaptive actor exploration, and scheduled critic exploration. Finally, we provide discretized updating rules and pseudocode for our ALM-RL algorithm.

3.1 Function Parameterization

While the direct application of the optimal solutions (7) from the continuous-time RL framework is impractical due to unknown model parameters, the structural insights guide our parameterization. Specifically, the optimal value function is quadratic in the surplus deviation x, and the mean of the optimal stochastic Gaussian policy is linearly dependent on x. Thus, we parameterize the value function with parameters $\boldsymbol{\theta} \in \mathbb{R}^d$:

$$J(t,x;\boldsymbol{\theta}) = -\frac{1}{2}k_1(t;\boldsymbol{\theta})x^2 + k_3(t;\boldsymbol{\theta}), \tag{8}$$

where both functions k_1 and k_3 are continuous in t and $\boldsymbol{\theta}$. And the policy with $\boldsymbol{\phi} = (\phi_1, \phi_2 > 0)^{\top}$, yielding a Gaussian distribution:

$$\pi(u \mid x;\boldsymbol{\phi}) = \mathcal{N}(u \mid \phi_1 x, \phi_2). \tag{9}$$

3.2 Policy Evaluation

Policy evaluation (PE) is a critical component in RL, focusing on learning the value function associated with a given control policy. Following the parameterization strategies outlined for the value function and policy in (8) and (9), PE involves updating the parameters $\boldsymbol{\theta}$ to refine the function approximations of $k_1(t;\boldsymbol{\theta})$ and $k_3(t;\boldsymbol{\theta})$. The Temporal Difference (TD) method proposed in [16] suggests an offline learning setting for updating $\boldsymbol{\theta}$ as follows:

$$\boldsymbol{\theta}_{n+1} \leftarrow \boldsymbol{\theta}_n + a_n \int_0^T \frac{\partial J}{\partial \boldsymbol{\theta}}(t, x_n(t);\boldsymbol{\theta}_n)\Big[\mathrm{d}J\left(t, x_n(t);\boldsymbol{\theta}_n\right) - \frac{1}{2}Q{x_n(t)}^2\mathrm{d}t + \gamma p\left(t,\boldsymbol{\phi}_n\right)\mathrm{d}t\Big], \tag{10}$$

where a_n denotes the learning rate, and the subscript n indicates the n-th episode throughout.

Furthermore, [14] theoretically proves that the convergence rate of policy parameters $\boldsymbol{\phi}$ is robust to the forms of $k_1(t;\boldsymbol{\theta})$ and $k_3(t;\boldsymbol{\theta})$, enabling flexible adaptations across different complexities in ALM modeling.

3.3 Policy Improvement

Policy improvement enhances the policy by iteratively updating the policy parameters based on feedback from the environment, aiming to increase expected performance. We adopt the continuous-time policy gradient (PG) method from [17] for ϕ_1. Moreover, to ensure stability when updating $\phi_{1,n}$, we need to consider the effect of diminishing exploration controlled by $\phi_{2,n}$. As $\phi_{2,n}$ becomes small,

the term $\phi_{2,n}^{-1}$ appearing in $\frac{\partial \log \pi}{\partial \phi_1}$ can lead to numerical instability. To address this, we multiply by $\phi_{2,n}$ during the update, effectively neutralizing the inverse and stabilizing the learning process.

$$\phi_{1,n+1} \leftarrow \phi_{1,n} + a_n Z_{1,n}(T), \tag{11}$$

where a_n is the learning rate and the term $Z_{1,n}(s)$ is defined as:

$$\begin{aligned} Z_{1,n}(s) = \int_0^s \frac{\partial \log \pi}{\partial \phi_1}\left(u_n(t) \mid x_n(t); \boldsymbol{\phi}_n\right) \Big[\mathrm{d}J(t, x_n(t); \boldsymbol{\theta}_n) \\ - \frac{1}{2} Q x_n(t)^2 \mathrm{d}t + \gamma p(t, \boldsymbol{\phi}_n) \mathrm{d}t \Big] \phi_{2,n}. \end{aligned} \tag{12}$$

3.4 Adaptive Actor Exploration

The actor's exploration level is governed by the variance of the stochastic policy, represented by ϕ_2. In the approach proposed by [14], $\phi_{2,n}$ follows a predetermined diminishing sequence, limiting the adaptability of exploration to evolving data.

To improve the adaptability of actor exploration, we employ the policy-gradient updating method from [15], which enables ϕ_2 to be updated dynamically in response to observed data. For computational efficiency in the stochastic approximation algorithm, we reparametrize ϕ_2 as ϕ_2^{-1}. By applying the chain rule, the derivative of ϕ_2^{-1} with respect to ϕ_2 simplifies to a time-invariant factor, which can be ignored in the gradient update, streamlining the computation. Consequently, we have

$$\phi_{2,n+1} \leftarrow \phi_{2,n} - a_n Z_{2,n}(T), \tag{13}$$

where a_n is the learning rate, and $Z_{2,n}(s)$ is defined as follows:

$$\begin{aligned} Z_{2,n}(s) = \int_0^s \Bigg\{ \frac{\partial \log \pi}{\partial \phi_{2,n}^{-1}}\left(u_n(t) \mid t, x_n(t); \boldsymbol{\phi}_n\right) \\ \left[\mathrm{d}J\left(t, x_n(t); \boldsymbol{\theta}_n\right) - \frac{1}{2} Q x_n(t)^2 \mathrm{d}t + \gamma p\left(t, \boldsymbol{\phi}_n\right) \mathrm{d}t \right] \\ + \gamma \frac{\partial p}{\partial \phi_{2,n}^{-1}}\left(t, \boldsymbol{\phi}_n\right) \mathrm{d}t \Bigg\}. \end{aligned} \tag{14}$$

3.5 Scheduled Critic Exploration

The temperature parameter γ plays a crucial role in RL by controlling the weight of the entropy-regularized term in the objective function, as outlined in (6). This parameter governs the level of exploration by the critic, influencing how much variability is incorporated into policy evaluations. A high γ value promotes exploration by emphasizing entropy, while a low γ value focuses on exploitation, which can lead to faster convergence but risks premature policy stagnation.

Maintaining a balance between exploration and exploitation is essential to ensure that the algorithm explores sufficiently during the early stages while converging effectively in later stages. To achieve this, γ needs to diminish over time, allowing the critic to gradually shift its focus from exploration to exploitation. Instead of using a fixed hyperparameter γ that requires extensive fine-tuning, as commonly done in continuous-time RL [16, 17], we propose a scheduled approach, where γ is defined as:

$$\gamma_n = \frac{c_\gamma}{b_n}, \qquad \text{for } n = 0, 1, \cdots \tag{15}$$

where c_γ is a constant that determines the exploration level, while $b_n > 1$ represents a monotone increasing sequence to infinity that governs the exploration scheduling. This formulation ensures that γ decreases systematically over time, providing a natural mechanism to balance exploration and exploitation without manual tuning.

3.6 Discretization and Projections

In our continuous-time RL framework, both the theoretical development and analysis are carried out entirely in continuous time. Discretization is introduced only at the final stage, solely for numerical implementation—specifically for approximating integrals and computing the $\mathrm{d}J$ term. To this end, the time interval $[0, T]$ is divided into uniform steps of length Δt. This final-stage discretization avoids the drawbacks of discretizing the problem at the outset (i.e., converting the continuous-time problem into a Markov Decision Process), which is known to cause performance instability when the timestep is small [23, 25, 33].

To ensure numerical stability during learning, we project the parameters onto convex sets:

$$\begin{aligned} &K_{\boldsymbol{\theta}} = \left\{ \boldsymbol{\theta} \in \mathbb{R}^d : |\boldsymbol{\theta}| \le U_{\boldsymbol{\theta}} \right\}, \qquad K_1 = \{\phi_1 \in \mathbb{R} : |\phi_1| \le U_1\}, \\ &K_2 = \{\phi_2 \in \mathbb{R} : \epsilon \le |\phi_2| \le U_2\}, \end{aligned}$$

where $U_{\boldsymbol{\theta}}$, U_1, and U_2 are fixed, sufficiently large positive constants that bound the parameter magnitudes. The constant $\epsilon > 0$ represents the minimum exploration level to enforce non-degenerate stochastic policies and can be chosen arbitrarily close to zero. In practice, these bounds can be tuned to improve empirical performance while preserving theoretical stability.

Finally, for convex set K, we define $\Pi_K(x) := \arg\min_{y \in K} |y - x|^2$. By employing a scheduled temperature γ_n and applying Euler discretization, the update rules for the parameters $\boldsymbol{\theta}_n$, $\phi_{1,n}$, and $\phi_{2,n}$ in (10), (11), and (13) are derived as follows:

$$\begin{aligned} \boldsymbol{\theta}_{n+1} \leftarrow \Pi_{K_{\boldsymbol{\theta}}}\Bigg(\boldsymbol{\theta}_n + a_n \sum_{k=0}^{\lfloor \frac{T}{\Delta t} - 1 \rfloor} \frac{\partial J}{\partial \boldsymbol{\theta}}(t_k, x_n(t_k); \boldsymbol{\theta}_n) \Big[-\frac{1}{2} Q x_n(t_k)^2 \Delta t \\ + \gamma_n p\left(t_k, \boldsymbol{\phi}_n\right) \Delta t + J\left(t_{k+1}, x_n(t_{k+1}); \boldsymbol{\theta}_n\right) - J\left(t_k, x_n(t_k); \boldsymbol{\theta}_n\right) \Big] \Bigg), \end{aligned} \tag{16}$$

$$\begin{aligned} \phi_{1,n+1} \leftarrow \Pi_{K_1}\Bigg(\phi_{1,n} + a_n \phi_{2,n} \sum_{k=0}^{\lfloor \frac{T}{\Delta t} - 1 \rfloor} \Bigg\{ \frac{\partial \log \pi}{\partial \phi_1}\left(u_n(t_k) \mid t_k, x_n(t_k); \boldsymbol{\phi}_n\right) \\ \Big[J\left(t_{k+1}, x_n(t_{k+1}); \boldsymbol{\theta}_n\right) - J\left(t_k, x_n(t_k); \boldsymbol{\theta}_n\right) - \frac{1}{2} Q x_n(t_k)^2 \Delta t \\ + \gamma_n p\left(t_k, \boldsymbol{\phi}_n\right) \Delta t \Big] + \gamma_n \frac{\partial p}{\partial \phi_1}\left(t_k, \boldsymbol{\phi}_n\right) \Delta t \Bigg\} \Bigg), \end{aligned} \tag{17}$$

$$\phi_{2,n+1} \leftarrow \Pi_{K_2}\Bigg(\phi_{2,n} - a_n \sum_{k=0}^{\lfloor \frac{T}{\Delta t}-1 \rfloor} \Bigg\{ \frac{\partial \log \pi}{\partial \phi_2^{-1}} \left(u_n(t_k) \mid t_k, x_n(t_k); \boldsymbol{\phi_n}\right) \Big[J\left(t_{k+1}, x_n(t_{k+1}); \boldsymbol{\theta_n}\right) - J\left(t_k, x_n(t_k); \boldsymbol{\theta_n}\right) - \frac{1}{2} Q x_n(t_k)^2 \Delta t + \gamma_n p\left(t_k, \boldsymbol{\phi_n}\right) \Delta t \Big] + \gamma_n \frac{\partial p}{\partial \phi_2^{-1}} \left(t_k, \boldsymbol{\phi_n}\right) \Delta t \Bigg\} \Bigg). \tag{18}$$

3.7 Pseudocode

Based on the analytical development presented, we outline the RL algorithm for the ALM problem as follows:

Algorithm 1 ALM-RL Algorithm

for $n = 1$ to N **do**
  Set $k = 0$, $t = t_k = 0$, $x_n(t_k) = x_0$
  while $t < T$ **do**
    Sample action $u_n(t_k)$ following stochastic policy (9)
    Update next surplus deviation $x_n(t_{k+1})$ using (1)
    Increment time: $t_{k+1} = t_k + \Delta t$
  end while
  Collect trajectory $\{(t_k, x_n(t_k), u_n(t_k))\}_{k\geq 0}$
  Update $\boldsymbol{\theta}$ and ϕ_1 via (16) and (17)
  Perform adaptive actor exploration using (18)
  Apply scheduled critic exploration via (15)
end for

4 Convergence Results

In this section, we present the convergence analysis for Algorithm 1. Throughout, we use c, and its variants, to denote generic positive constants that may vary from line to line. These constants depend only on the model parameters A, B, C, D, Q, H, the initial condition x_0, time horizon T, and the predefined algorithmic hyperparameters $c_\gamma, U_{\boldsymbol{\theta}}, U_1, U_2$, and ϵ.

Theorem 1. *Suppose the learning rate sequence $\{a_n\}$ satisfies the standard conditions:*

$$\sum a_n = \infty, \quad \sum a_n^2 < \infty. \tag{19}$$

Then, the following almost surely convergence holds:

$$\phi_{1,n} \xrightarrow{a.s.} \phi_1^* = -\frac{B + CD}{D^2},$$

and

$$\phi_{2,n} \xrightarrow{a.s.} \epsilon.$$

Remark 1. *ϕ_1^* represents the oracle value, corresponding to the explicit solution under the assumption of complete market knowledge; see Equations* (3) *and* (7) *for the optimal value $\phi_1^* = -\frac{B+CD}{D^2}$.*

Proof. The proof strategy builds on the framework established in [14, Theorem 4.1] and [15, Theorem 5.1], with necessary adaptations to the current Algorithm 1 under ALM setting. It also leverages classical results from stochastic approximation theory [1, 27, 28].

Firstly, we denote the mean part $h_1(\phi_{1,n}, \phi_{2,n}; \boldsymbol{\theta}_n) = \mathbb{E}[Z_{1,n}(T) \mid \boldsymbol{\theta}_n, \boldsymbol{\phi}_n]$ and noise part $\xi_{1,n} = Z_{1,n}(T) - h_1(\phi_{1,n}, \phi_{2,n}; \boldsymbol{\theta}_n)$, so that the updating rule for ϕ_1 is

$$\phi_{1,n+1} = \Pi_{K_{1,n+1}}(\phi_{1,n} + a_n[h_1(\phi_{1,n}, \phi_{2,n}; \boldsymbol{\theta}_n) + \xi_{1,n}]). \tag{20}$$

Applying Ito's lemma to the process $J(t, x_n(t); \boldsymbol{\theta}_n)$ then to $Z_{1,n}$, we have

$$\begin{aligned} \mathrm{d}Z_{1,n}(t) = (u_n(t) - \phi_{1,n} x_n(t)) x_n(t) \Bigg\{ \Bigg[& -\frac{1}{2} k_1'(t; \boldsymbol{\theta}_n) x_n(t)^2 + k_3'(t; \boldsymbol{\theta}_n) - (A x_n(t) + B u_n(t)) k_1(t; \boldsymbol{\theta}_n) x_n(t) - \frac{1}{2} Q x_n(t)^2 \\ & - \frac{(C x_n(t) + D u_n(t))^2}{2} k_1(t; \boldsymbol{\theta}_n) + \frac{\gamma}{2} \log(2\pi e \phi_{2,n}) \Bigg] \mathrm{d}t \\ & - \Big((C x_n(t) + D u_n(t)) k_1(t; \boldsymbol{\theta}_n) x_n(t) \Big) \mathrm{d}W_n(t) \Bigg\}. \end{aligned} \tag{21}$$

Then by [14, Lemma B.1], we can get the noise bound

$$\begin{aligned} & \mathrm{Var}\left(\xi_{1,n} \middle| \boldsymbol{\theta}_n, \phi_{1,n}, \phi_{2,n}\right) \\ \leq & c' \left(1 + |\phi_{1,n}|^8 + (\log \phi_{2,n})^8\right) \exp\{c' |\phi_{1,n}|^6\} \\ \leq & c' \left(1 + U_1^8 + (\log U_2)^8 + (\log \epsilon)^8\right) \exp\{c' U_1^6\} \leq c, \end{aligned} \tag{22}$$

and the mean part

$$h_1(\phi_{1,n}, \phi_{2,n}; \boldsymbol{\theta}_n) = -l(\phi_{1,n}, \phi_{2,n}; \boldsymbol{\theta}_n)(\phi_{1,n} - \phi_1^*), \tag{23}$$

where

$$l(\phi_{1,n}, \phi_{2,n}; \boldsymbol{\theta}_n) = D^2 \phi_{2,n} \int_0^T k_1(t; \boldsymbol{\theta}_n) \mathbb{E}[x_n(t)^2] \mathrm{d}t. \tag{24}$$

Moreover, we can further derive that $l(\phi_{1,n}, \phi_{2,n}; \boldsymbol{\theta}_n) \geq \bar{c} > 0$ and $|h_1(\phi_{1,n}, \phi_{2,n}; \boldsymbol{\theta}_n)| \leq c' U_2 (1 + U_1) e^{c' U_1^2} \leq c$.

Next, we let $\{\mathcal{G}_n\}$ be the filtration generated by $\{\boldsymbol{\theta}_m, \phi_{1,m}, \phi_{2,m}, m = 0, 1, 2, ..., n\}$ and denote $U_{1,n} = \phi_{1,n} - \phi_1^*$. Then we have

$$\begin{aligned} & \mathbb{E}\left[|U_{1,n+1}|^2 \middle| \mathcal{G}_n \right] \\ \leq & \mathbb{E}\left[|U_{1,n} + a_n[h_1(\phi_{1,n}, \phi_{2,n}; \boldsymbol{\theta}_n) + \xi_{1,n}]|^2 \middle| \mathcal{G}_n \right] \\ \leq & |U_{1,n}|^2 + 2 a_n U_{1,n} h_1(\phi_{1,n}, \phi_{2,n}; \boldsymbol{\theta}_n) + \\ & + 3 a_n^2 \left(|h_1(\phi_{1,n}, \phi_{2,n}; \boldsymbol{\theta}_n)|^2 + \mathbb{E}\left[\left|\xi_{1,n}\right|^2 \middle| \mathcal{G}_n \right] \right) \\ \leq & |U_{1,n}|^2 + 2 a_n U_{1,n} h_1(\phi_{1,n}, \phi_{2,n}; \boldsymbol{\theta}_n) + c a_n^2. \end{aligned}$$

Following from [28, Theorem 1], we know that $\left|U_{1,n}\right|^2$ converges to a finite limit and $\sum -a_n U_{1,n} h_1(\phi_{1,n}, \phi_{2,n}; \boldsymbol{\theta}_n) < \infty$ almost surely. Then by (23), (24) and lower bound of $l(\phi_{1,n}, \phi_{2,n}; \boldsymbol{\theta}_n)$,

$$-a_n U_{1,n} h_1(\phi_{1,n}, \phi_{2,n}; \boldsymbol{\theta}_n) = 2 a_n l(\phi_{1,n}, \phi_{2,n}; \boldsymbol{\theta}_n) U_{1,n}^2 \geq 2\bar{c} a_n U_{1,n}^2.$$

To prove $U_{1,n}^2 \to 0$, we suppose $U_{1,n}^2 \to r$ almost surely, where $0 < c < \infty$ is a constant. Then there exists an n_0 and $0 < \delta < r$ such that $U_{1,n}^2 \geq r - \delta > 0$ for $n > n_0$. Thus, by the assumption of this theorem, we have

$$\sum -a_n U_{1,n} h_1(\phi_{1,n}, \phi_{2,n}; \boldsymbol{\theta}_n) \geq \sum 2\bar{c} a_n U_{1,n}^2 \geq \sum 2\bar{c} a_n (r - \delta) = \infty,$$

which contradicts with $\sum -a_n U_{1,n} h_1(\phi_{1,n}, \phi_{2,n}; \boldsymbol{\theta}_n) < \infty$. Therefore, $\phi_{1,n}$ converges to ϕ_1^* almost surely. The almost sure convergence of $\phi_{2,n}$ to ϵ follows from a similar argument as $\phi_{1,n}$. □

This proof is included for completeness and builds on [14, 15], with key differences in the use of uniform bounds and a minimum exploration level specific to ALM.

5 Numerical Experiments

This section details simulation experiments that compare our ALM-RL algorithm against six alternative strategies. The comparisons include two enhanced traditional financial methods, one model-based continuous-time RL strategy, and three established RL algorithms, each described in the subsequent subsection.

5.1 Comparative ALM Strategies

5.1.1 Dynamic CPPI Strategy. To ensure comparability with our ALM-RL algorithm and other RL methods, the Dynamic Constant Proportion Portfolio Insurance (DCPPI) strategy incorporates an adaptive multiplier, traditionally constant in CPPI [4, 9]. This adjustment enhances the strategy's ability to learn and adapt, overcoming the traditional CPPI's limitation where the performance is highly dependent on the initially chosen multiplier m.

DCPPI seeks to maintain zero deviation between the current surplus and the target surplus by dynamically adjusting to changing market conditions. The policy u is defined as:

$$u^{DCPPI}(t) = -m \cdot x(t), \tag{25}$$

where m is adaptively updated using a data-driven approach. Starting from an initial value m_0, we simulate a trajectory of surplus deviation $x_0, x_1, ..., x_l$, and adjust m based on the directionality of changes between consecutive surplus deviations:

$$m_{n+1} = m_n + a_n \cdot \operatorname{sgn}\left(\sum_{i=0}^{l-1} \operatorname{sgn}(x_i \cdot x_{i+1})\right), \tag{26}$$

where a_n is the learning rate and $\operatorname{sgn}(\cdot)$ is the sign function. This updating rule ensures that m is modified to correct the previous trajectory's trend by considering the sign consistency between consecutive surplus deviations, enhancing the responsiveness and accuracy of the strategy in aligning with market dynamics.

5.1.2 Adaptive Contingent Strategy. Drawing on the principles of contingent immunization [18, 19], the Adaptive Contingent Strategy (ACS) aims to maintain the surplus deviation within predefined tolerance levels (δ), contrasting with the DCPPI's zero-deviation from the target. This conservative approach allows for minor fluctuations within a safe boundary and opts for inaction when the surplus deviation is adequately balanced, thereby avoiding unnecessary market exposure and reducing noise amplification from market volatility. This is particularly useful given the stochastic nature of financial movements, specifically the volatility component $Du(t)\mathrm{d}W(t)$ in the ALM dynamics (1).

The policy u is formulated to be minimally interfered:

$$u^{ACS}(t) = -m \cdot \operatorname{sgn}(x(t)) \cdot \max(|x(t)| - \delta, 0), \tag{27}$$

where the multiplier m dynamically updates similarly to (26).

5.1.3 Model-Based Plugin Strategy. The Model-Based Plugin Strategy (MBP), derived from the continuous-time RL algorithm by [3, 32], primarily estimates parameters A and B under assumptions of constant volatility, and then plugs these estimates into analytical solutions. This algorithm has been mathematically proven to offer fast convergence. To align with the dynamic complexities of the ALM problem, which involves state- and control-dependent volatilities, this approach has been extended to also estimate parameters C and D using least squares regression, as detailed in [14]. This extension provides a clear contrast to our model-free, continuous-time RL approach.

5.1.4 Advanced RL Strategies. In our comparative analysis, we include three prominent RL algorithms—Soft Actor-Critic (SAC) [8], Proximal Policy Optimization (PPO)[29], and Deep Deterministic Policy Gradient (DDPG) [21]—due to their distinct characteristics and relevance in advancing RL applications. SAC is selected for its entropy-enhanced exploration technique which aligns with our model's entropy-based approach, emphasizing efficient exploration in continuous action spaces. PPO is included as a state-of-the-art representative for its ability to ensure stable and reliable policy updates, which is crucial for consistent performance across diverse market conditions. DDPG is chosen as a commonly referenced benchmark in RL studies, known for its foundational role in integrating deterministic policy gradient concepts with deep learning frameworks.

5.2 Experiment Setup

To evaluate the performance of our method, we conduct simulations designed to reflect realistic and uncertain ALM scenarios. While most existing studies in the ALM literature [6, 20, 40, 42] evaluate algorithms under fixed model parameters, we adopt a randomized setup to better reflect the lack of prior knowledge in real-world financial markets and to assess the algorithm's robustness across diverse environments. The parameter ranges are chosen based on typical values used in these studies and general financial intuition: $A \sim \mathcal{U}(-0.05, 0.05)$, $B \sim \mathcal{U}(0.05, 0.15)$, and $C, D \sim \mathcal{U}(0.1, 0.2)$. Each simulation runs for 20000 episodes with a discretization step size of $\Delta t = 0.01$, and is repeated independently 200 times to ensure statistically reliable results.

Furthermore, the learning rate $a_n = (n+1)^{-3/4}$, shown to be effective in [14], is used for ALM-RL as well as for the two enhanced traditional ALM methods, DCPPI and ACS. This choice satisfies the standard assumption in (19), which is also the only assumption required in Theorem 1. For ALM-RL, the exploration scheduling sequence $b_n = (n+1)^{1/4}$ is additionally employed to accelerate convergence, as demonstrated in [14], from which both a_n and b_n are adopted. The initial actor exploration level is set to $\phi_{2,0} = 1$, and the constant $c_\gamma = 1$. The projection bounds are set to $U_{\boldsymbol{\theta}} = U_1 = U_2 = 100$, and the minimum exploration level is $\epsilon = 0.01$. The tolerance level for ACS is set as $\delta = 0.1$. Finally, all other learnable parameters across all ALM strategies are initialized using standard normal distributions.

The settings for MBP follow those in [3, 14, 32]. For SAC, PPO, and DDPG, the neural networks (NNs) used for both the actor and critic are feedforward architectures with two hidden layers, each containing 32 neurons and ReLU activation. Other hyperparameters

are mostly adopted from [8, 21, 29]. Lastly, to ensure reproducibility and fair comparisons under the randomized environment, we use 200 different random seeds to represent 200 independent market scenarios. Each method is evaluated under the same set of seeds so that all strategies face identical market conditions in each run.

5.3 Evaluation Metric

The average reward, a commonly used metric in RL to assess performance, is employed to compare the effectiveness of our ALM-RL algorithm with six alternative strategies. Following the value function (2), for each independent experiment, the reward is computed as:

$$\text{Reward} = \sum_{k=0}^{\lfloor \frac{T}{\Delta t}-1 \rfloor} -\frac{1}{2} Q \left(x_n(t_k)\right)^2 \Delta t - \frac{1}{2} H \left(x_n(T)\right)^2, \tag{28}$$

where $x_n(t_k)$ represents the surplus deviation at time step t_k.

For each method, the average reward per episode is computed as the mean of rewards from 200 independent experiments, resulting in an average reward curve over 20,000 episodes. This curve provides a reliable measure for comparing the methods' performance, learning dynamics, and overall effectiveness.

5.4 Performance Evaluation of ALM Strategies

Now we analyze the performance of ALM strategies in randomized market conditions across 200 independent runs, as illustrated in Figure 1.

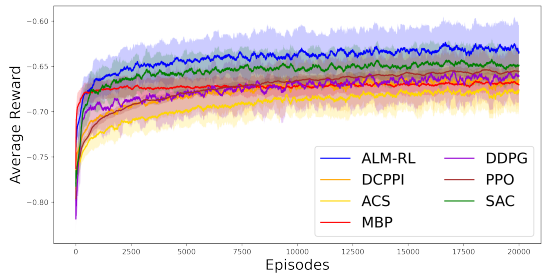

Figure 1: Average reward under randomized market parameters, smoothed with a 200-point moving average over 20000 episodes. Th shaded area indicates the interquartile range, based on 200 independent simulations.

From Figure 1, we see that our proposed ALM-RL algorithm consistently outperforms all other strategies across almost all episodes. It exhibits rapid initial gains and sustained superiority throughout the learning horizon, demonstrating notable resilience and adaptability in refining its strategy more effectively than competing methods. This strong performance is likely due to its use of entropy-regularization exploration techniques and non-degenerate stochastic policies, which enhance adaptability and decision-making under uncertainty. Similarly, SAC, which also leverages entropy techniques and stochastic policies, achieves rapid initial gains and maintains the second-best performance after 2500 episodes.

PPO, due to its conservative clipped surrogate objective, starts off slower but gradually approaches SAC's performance, albeit remaining slightly lower. The clipping mechanism yields the smoothest learning curve and the narrowest interquartile range (IQR) among all strategies, reflecting high stability across runs. Compared with SAC and PPO, DDPG exhibits moderate initial growth but ultimately settles at a lower performance level. It also exhibits the highest volatility and widest IQR, likely due to the sensitivity of its deterministic policy gradient to noise and outliers. The MBP strategy demonstrates rapid early growth and attains a relatively high average reward in the initial stage. However, its performance soon stagnates, converging to suboptimal solutions. This lack of continued improvement is reflected in the flat reward curve and is likely attributable to parameter estimation errors inherent in financial markets [22]. Finally, DCPPI and ACS achieve one of the lowest terminal rewards but maintain reliable, smooth performance throughout. Moreover, DCPPI's proactive control yields better outcomes than the more conservative ACS, reinforcing the importance of active management.

Moreover, in order to show statistical significance, we conduct one-sided Wilcoxon paired tests between each pair of ALM strategies, and the resulting matrix of p-values is presented in Figure 2. Each cell displays the p-value for the null hypothesis that the row method does not outperform the column method. Darker shades indicate stronger statistical evidence against the null. Notably, ALM-RL demonstrates statistically significant improvements over all strategies except SAC at the 95% confidence level, and over SAC at the 90% level, with corresponding p-values below 0.05 and 0.10, respectively. This supports the robustness and consistent superiority of ALM-RL across randomized environments.

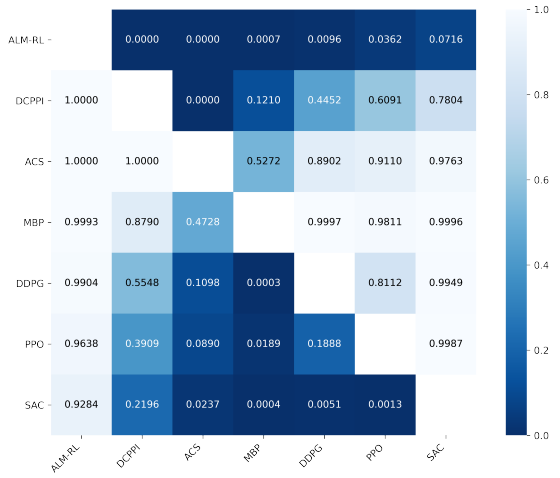

Figure 2: Heatmap of p-values from one-sided Wilcoxon paired tests comparing the terminal reward of each ALM strategy. The terminal reward is calculated as the average reward over the last 500 episodes to reduce the noise that may result from using a single final episode.

6 Conclusions

This paper introduced a novel approach for ALM by formulating the problem as a stochastic LQ control and solving it within a model-free, continuous-time RL framework. In addition to the new formulation, we integrated adaptive exploration for the actor and scheduled exploration for the critic, ensuring an effective exploration and exploitation trade-off. Notably, despite the use of adaptive and scheduled exploration techniques, we are able to prove almost sure convergence of all policy parameters. Our policy gradient-based soft actor-critic method was evaluated against two enhanced traditional financial strategies, a model-based continuous-time RL approach, and state-of-the-art RL algorithms, including SAC, PPO, and DDPG. The results consistently demonstrate that our method outperforms these alternatives across diverse market conditions.

The superior performance results from directly learning optimal ALM strategies without assuming any knowledge of the financial environment or estimating market parameters, highlighting a fundamental advantage of our approach. Future research will focus on extending this framework to broader financial domains and evaluating its performance in more complex and dynamic market environments.

References

[1] Sigrún Andradóttir. 1995. A stochastic approximation algorithm with varying bounds. *Operations Research* 43, 6 (1995), 1037–1048.

[2] Leemon C Baird. 1994. Reinforcement learning in continuous time: Advantage updating. In *Proceedings of 1994 IEEE International Conference on Neural Networks (ICNN'94)*, Vol. 4. IEEE, 2448–2453.

[3] Matteo Basei, Xin Guo, Anran Hu, and Yufei Zhang. 2022. Logarithmic regret for episodic continuous-time linear-quadratic reinforcement learning over a finite-time horizon. *Journal of Machine Learning Research* 23, 178 (2022), 1–34.

[4] Fischer Black and AndreF Perold. 1992. Theory of constant proportion portfolio insurance. *Journal of Economic Dynamics and Control* 16, 3-4 (1992), 403–426.

[5] Shuping Chen, Xunjing Li, and Xun Yu Zhou. 1998. Stochastic linear quadratic regulators with indefinite control weight costs. *SIAM Journal on Control and Optimization* 36, 5 (1998), 1685–1702.

[6] Mei Choi Chiu and Hoi Ying Wong. 2012. Mean–variance asset–liability management: Cointegrated assets and insurance liability. *European Journal of Operational Research* 223, 3 (2012), 785–793.

[7] Kenji Doya. 2000. Reinforcement learning in continuous time and space. *Neural Computation* 12, 1 (2000), 219–245.

[8] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. 2018. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International Conference on Machine Learning*. PMLR, 1861–1870.

[9] Erol Hakanoglu, Robert Kopprasch, and Emmanuel Roman. 1989. Constant proportion portfolio insurance for fixed-income investment. *Journal of Portfolio Management* 15, 4 (1989), 58.

[10] Thomas SY Ho. 1992. Key rate durations: Measures of interest rate risks. *The Journal of Fixed Income* 2, 2 (1992), 29–44.

[11] Ying Hu, Xiaomin Shi, and Zuo Quan Xu. 2022. Non-homogeneous stochastic LQ control with regime switching and random coefficients. *arXiv preprint arXiv:2201.01433* (2022).

[12] Yilie Huang, Yanwei Jia, and Xunyu Zhou. 2022. Achieving mean–variance efficiency by continuous-time reinforcement learning. In *Proceedings of the Third ACM International Conference on AI in Finance*. 377–385.

[13] Yilie Huang, Yanwei Jia, and Xun Yu Zhou. 2024. Mean–Variance Portfolio Selection by Continuous-Time Reinforcement Learning: Algorithms, Regret Analysis, and Empirical Study. *arXiv preprint arXiv:2412.16175* (2024).

[14] Yilie Huang, Yanwei Jia, and Xun Yu Zhou. 2025. Sublinear regret for a class of continuous-time linear-quadratic reinforcement learning problems. *SIAM Journal on Control and Optimization* 63, 5 (2025), 3452–3474.

[15] Yilie Huang and Xun Yu Zhou. 2025. Data-Driven Exploration for a Class of Continuous-Time Linear–Quadratic Reinforcement Learning Problems. *arXiv preprint arXiv:2507.00358* (2025).

[16] Yanwei Jia and Xun Yu Zhou. 2022. Policy evaluation and temporal-difference learning in continuous time and space: A martingale approach. *Journal of Machine Learning Research* 23, 154 (2022), 1–55.

[17] Yanwei Jia and Xun Yu Zhou. 2022. Policy gradient and actor-critic learning in continuous time and space: Theory and algorithms. *Journal of Machine Learning Research* 23, 154 (2022), 1–55.

[18] Martin L Leibowitz and Alfred Weinberger. 1982. Contingent immunization—Part I: Risk control procedures. *Financial Analysts Journal* 38, 6 (1982), 17–31.

[19] Martin L Leibowitz and Alfred Weinberger. 1983. Contingent Immunization—Part II: Problem Areas. *Financial Analysts Journal* 39, 1 (1983), 35–50.

[20] Chanjuan Li, Zhongfei Li, Ke Fu, and Haiqing Song. 2013. Time-consistent optimal portfolio strategy for asset-liability management under mean-variance criterion. *Accounting and Finance Research* 2, 2 (2013), 1–89.

[21] Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. 2015. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971* (2015).

[22] David G Luenberger. 1998. *Investment Science*. Oxford University Press.

[23] Rémi Munos. 2006. Policy gradient in continuous time. *Journal of Machine Learning Research* 7 (2006), 771–791.

[24] Jian Pan, Zujin Zhang, and Xiangying Zhou. 2018. Optimal dynamic mean-variance asset-liability management under the Heston model. *Advances in Difference Equations* 2018 (2018), 1–16.

[25] Seohong Park, Jaekyeom Kim, and Gunhee Kim. 2021. Time discretization-invariant safe action repetition for policy gradient methods. *Advances in Neural Information Processing Systems* 34 (2021), 267–279.

[26] MA Rami and Xun Yu Zhou. 2000. Linear matrix inequalities, Riccati equations, and indefinite stochastic linear quadratic controls. *IEEE Trans. Automat. Control* 45, 6 (2000), 1131–1143.

[27] Herbert Robbins and Sutton Monro. 1951. A stochastic approximation method. *The Annals of Mathematical Statistics* (1951), 400–407.

[28] Herbert Robbins and David Siegmund. 1971. A convergence theorem for non negative almost supermartingales and some applications. In *Optimizing Methods in Statistics*. Elsevier, 233–257.

[29] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).

[30] Yang Shen, Jiaqin Wei, and Qian Zhao. 2020. Mean–variance asset–liability management problem under non-Markovian regime-switching models. *Applied Mathematics & Optimization* 81 (2020), 859–897.

[31] Richard S Sutton and Andrew G Barto. 2018. *Reinforcement learning: An introduction*. MIT press.

[32] Lukasz Szpruch, Tanut Treetanthiploet, and Yufei Zhang. 2024. Optimal Scheduling of Entropy Regularizer for Continuous-Time Linear-Quadratic Reinforcement Learning. *SIAM Journal on Control and Optimization* 62, 1 (2024), 135–166.

[33] Corentin Tallec, Léonard Blier, and Yann Ollivier. 2019. Making deep Q-learning methods robust to time discretization. In *International Conference on Machine Learning*. PMLR, 6096–6104.

[34] Wenpin Tang and Xun Yu Zhou. 2024. Regret of exploratory policy improvement and q-learning. *arXiv preprint arXiv:2411.01302* (2024).

[35] Kyriakos G Vamvoudakis and Frank L Lewis. 2010. Online actor–critic algorithm to solve the continuous-time infinite horizon optimal control problem. *Automatica* 46, 5 (2010), 878–888.

[36] Robert Van der Meer and Meye Smink. 1993. Strategies and techniques for asset-liability management: an overview. *Geneva Papers on Risk and Insurance. Issues and Practice* (1993), 144–157.

[37] Haoran Wang, Thaleia Zariphopoulou, and Xun Yu Zhou. 2020. Reinforcement learning in continuous time and space: A stochastic control approach. *Journal of Machine Learning Research* 21, 198 (2020), 1–34.

[38] Xiaoli Wei and Xiang Yu. 2023. Continuous-time q-learning for McKean-Vlasov control problems. *arXiv preprint arXiv:2306.16208* (2023).

[39] AJ Wise. 1984. The matching of assets to liabilities. *Journal of the Institute of Actuaries* 111, 3 (1984), 445–501.

[40] Haixiang Yao, Yongzeng Lai, and Yong Li. 2013. Continuous-time mean–variance asset–liability management with endogenous liabilities. *Insurance: Mathematics and Economics* 52, 1 (2013), 6–17.

[41] Jiongmin Yong and Xun Yu Zhou. 1999. *Stochastic Controls: Hamiltonian Systems and HJB Equations*. New York, NY: Spinger.

[42] Miao Zhang and Ping Chen. 2016. Mean–variance asset–liability management under constant elasticity of variance process. *Insurance: Mathematics and Economics* 70 (2016), 11–18.

[43] Xun Yu Zhou and Duan Li. 2000. Continuous-time mean-variance portfolio selection: A stochastic LQ framework. *Applied Mathematics and Optimization* 42, 1 (2000), 19–33.

[44] Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, and Anind K Dey. 2008. Maximum entropy inverse reinforcement learning.. In *AAAI*, Vol. 8. Chicago, IL, USA, 1433–1438.