Report for Corporate Finance: Part I

Wenhao ZHANG

January 1, 2026

Contents

1 Introduction

Dynamic corporate finance studies how firms make sequential decisions on investment, financing, payout, etc., depending on the settings of models. In this report we will explain and discuss part I of question 4, which focuses on applying deep learning method to dynamic corporate finance model.

In the area of dynamic corporate finance, the firm is modeled as an intertemporal decision maker that maximizes shareholder value in the presence of uncertainty, financing frictions, adjustment costs, etc. [6, 7]. Unlike static Modigliani–Miller benchmarks [2], dynamic models emphasize path dependence, optimal stopping (default), and endogenous state evolution (capital, debt, cash, productivity shocks, belief states, etc.) [3, 5]. Intertemporal optimization perspective of corporate decision making can be traced back to the recursive dynamic framework in [1]. The first stochastic dynamic model that treats corporate default as an endogenous optimal stopping problem is introduced by Merton [3], marking the earliest structural foundation aligned with the spirit of dynamic corporate finance. The first explicit dynamic optimal capital structure model incorporating tax shields, bankruptcy costs, and maturity structure is developed by Leland [4] and extended by Leland and Toft [5], forming the most direct prototype of modern dynamic corporate finance models.

It is worth noting that this idea of dynamic decision-making has been widely studied in optimal control, financial mathematics, and reinforcement learning, and has achieved many important results. For instance, Wang and Zhou [9] study the mean–variance portfolio selection problem, while Dai et al. [10] investigate Merton's expected utility maximization problem in incomplete markets. Both works adopt model-free reinforcement learning methods.

Given that most models rely on Markov dynamics, we may formulate the discrete-time dynamic corporate finance problem as the following optimal control problem. (Extensions to continuous time and finite decision horizons require slight adjustments)

$$\begin{aligned} \max_{u(t,x)} \quad & \mathbb{E}\left[\sum_{t=0}^{\infty} r(t, x_t, u_t)\right] \\ \text{s.t.} \quad & x_{t+1} = f(t, x_t, u_t), \\ & u_t = u(t,x) \in \mathcal{U}(x_t), \\ & x_t \in \mathcal{X}. \end{aligned}$$

where x_t denotes the *state*, summarizing all accessible relevant information at time t. $\mathcal{X}$ denotes the *state space*, i.e., the set of all feasible states. u_t denotes the *control variable* at time t, generated by a control function $u(t,x)$. $\mathcal{U}(x)$ denotes the *admissible control set* at state x. $f(t,x,u)$ denotes the system dynamics that maps the current state x_t and control u_t into the next state x_{t+1} (possibly be stochastic).

The two dynamic corporate finance problems considered in this report are both special cases of the general infinite-horizon discrete-time optimal control formulation described above.

2 Basic model

Basic model in [6] can be rewritten as:

$$\max_{\{I_t\}_{t=0}^{\infty}} \mathbb{E}\Bigg[\sum_{t=0}^{\infty} \beta^t e(k_t, I_t, z_t)\Bigg]$$

subject to:

$$k_{t+1} = (1-\delta)k_t + I_t$$
$$\ln z_{t+1} = \rho \ln z_t + \epsilon_t$$

where $\epsilon_t \sim TN(0, \sigma_\epsilon; a, b)$, $e(k_t, I_t, z_t) = \pi(k_t, z_t) - \psi(I_t, k_t) - I_t$, $\beta \in (0,1)$ is a discount factor, which is equivalent to $1/(1+r)$ in [6]. In this case, the state variable is (k_t, z_t) and the control variable is I_t. From this point onward, whenever a letter that also appears in [6] is used in this paper, it carries the same meaning. For example, k_t denotes the beginning-of-period capital stock, $\pi(k,t)$ is the profit function, I_t represents investment in capital, etc.

2.1 Deep learning method to solve basic model

To use deep learning method in [8], we need to construct the following objectives. The first, and the most straightforward approach is to perform direct maximization to the objective functional, i.e.,

$$\max_{\Theta} \Xi\Bigg[\sum_{t=0}^{\infty} \beta^t e(k_t, I^{\Theta}(k_t, z_t), z_t)\Bigg] \tag{1}$$

where $\Xi[\cdot]$ denotes the sample expectation. $I^{\Theta}(k_t, z_t)$ denotes deep neutral network (DNN) with input (k_t, z_t) and parameter Θ.

The second approach in [8] is to minimize the Euler-residual. The Lagrangian is defined as:

$$\mathcal{L} = \mathbb{E}\Bigg[\sum_{t=0}^{\infty} \beta^t \big(e(k_t, I_t, z_t) - \lambda_t(k_{t+1} - (1-\delta)k_t - I_t\big)\Bigg].$$

Using first order condition we have:

$$\frac{\partial \mathcal{L}}{\partial I_t} = e_I(k_t, I_t, z_t) + \lambda_t = 0,$$
$$\frac{\partial \mathcal{L}}{\partial k_{t+1}} = \beta \mathbb{E}\big[e_k(k_{t+1}, I_{t+1}, z_{t+1}) + (1-\delta)\lambda_{t+1}\big] - \lambda_t = 0$$

where $e_I(k_t, I_t, z_t) = -\psi_I(I_t, k_t) - 1, e_k(k_{t+1}, I_{t+1}, z_{t+1}) = \pi_k(k_{t+1}, z_{t+1}) - \psi_k(I_{t+1}, k_{t+1})$. The Euler equation will be

$$\psi_I(I_t, k_t) + 1 = \beta\mathbb{E}\big[\pi_k(k_{t+1}, z_{t+1}) - \psi_k(I_{t+1}, k_{t+1}) + (1-\delta)(\psi_I(I_{t+1}, k_{t+1}) + 1)\big].$$

So the second approach is to

$$\min_{\Theta} \Xi\Big[\psi_I(I^\Theta(k_{t+1}, z_{t+1}), k_t) - \beta\big[\pi_k(k_{t+1}, z_{t+1}) - \psi_k(I^\Theta(k_t, z_t), k_{t+1}) + (1-\delta)(\psi_I(I^\Theta(k_{t+1}, z_{t+1}), k_{t+1}))\big]\Big]^2 \tag{2}$$

The third approach is to minimize the Bellman-residual. We first derive Bellmen equation using dynamic programming,

$$V(k, z) = \max_I \big\{e(k, I, z) + \beta\mathbb{E}_{z'}V((1-\delta)k + I, z')\big\}.$$

So the third approach is to

$$\min_{\Theta,\Phi} \Xi\ \big[V^\Phi(k, z) - \max_I \big\{e(k, I, z) + \beta V((1-\delta)k + I, z_{t+1})\big\}\big]^2 \tag{3}$$

where $V^\Phi(k, z)$ denotes a DNN with input (k, z) and parameter Φ.

To sum up, we have formulated the following DL framework to solve the basic model in 3 approaches

Approach	Lifetime-reward	Euler-residual	Bellman-residual
Objective function	(1)	(2)	(3)
DNN	$I^\Theta(k, z)$	$I^\Theta(k, z)$	$I^\Theta(k, z)$ and $V^\Phi(k, z)$

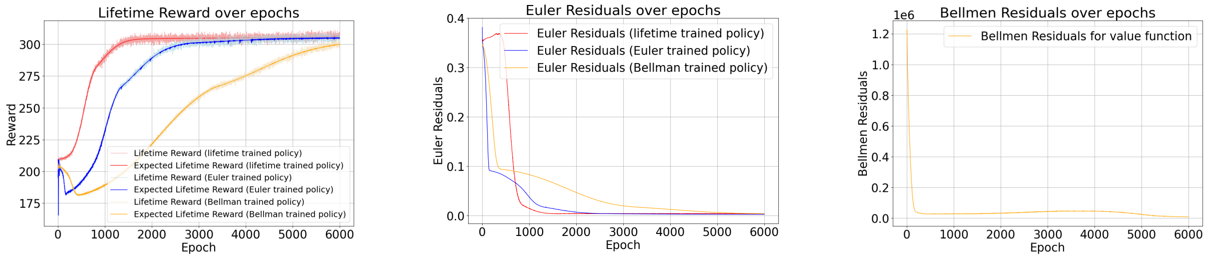
Table 1: empirical risk construction

2.2 Approaches to test effectiveness

Explicit solutions for stochastic control problems are generally unavailable, especially in discrete-time settings, which complicates the verification of solution optimality. In most cases, we can only observe whether our algorithm causes the value function to converge to a stable and relatively high value, and compared with some benchmark, if possible.

In this problem, the first and most straightforward approach is to observe whether the lifetime reward converges to a high value. Furthermore, since we employ three methods, determining whether they all converge to the same reward level is a crucial indicator. The lifetime rewards for these three methods are shown in the figure 1a.

It is well known that the first-order condition (FOC) is a necessary condition for optimality. Therefore, our second approach is to observe whether the derivative of $\partial\mathcal{L}/\partial I_t$ converges to zero, i.e., whether $\sum_{t=0}^{T}(\partial\mathcal{L}/\partial I_t)^2$ converges to 0. This aligns with Euler-residual method in (2). during training process. Figure 1b verifies effectiveness of three methods. Besides In addition, as shown in Figure 1c, the third method incorporates an extra value function network, which can be used to predict $V(k, z)$ in practice.



(a) Life-time reward for three methods over training epochs

(b) $\sum_{t=0}^{T}(\partial\mathcal{L}/\partial I_t)^2$ for three methods over training epochs

(c) Prediction error for value function

Figure 1: Effective metrics

In this problem, the system is relatively simple and can generate an infinite amount of data based on the specific functional forms; thus, overfitting is not a concern. However, to extend the model to broader or even practical applications in the future, we must ensure its robustness. Therefore, we propose third validation method: cross-validation. This is a method widely utilized in statistics and supervised learning, yet it remains under-explored in the field of dynamic corporate finance. Specifically, we can employ K-fold cross-validation to examine whether our model possesses generalization capabilities with respect to initial values or even system dynamics.

2.3 Issues to consider

During the process of completing the report, I encountered many challenges. I must admit that the experience of identifying and solving problems is truly fascinating. Although most issues are effectively resolved, some remain unsolved due to limited time. Below, I provide a point-by-point explanation, covering both resolved and unresolved parts.

2.3.1 Non-convex optimization landscape

Consider the previous basic model, we design a DNN with parameter Θ and input (k, z). Thanks to stochastic optimization techniques (e.g., Adam and SGD) and parallel computing, we can get $\Theta^* \approx \inf_\Theta \ell$ in most cases efficiently, where ℓ denotes the loss function. However, in our basic model, optimization of DNN is not that trivial. In the beginning of implementing, I encounter gradient vanishing and exploding in many times. In my research, I find that this issue is mainly caused by two factors. The first is that we study an infinite-time-horizon problem. In control theory, solvability and stability are fundamental challenges that cannot be avoided in infinite-horizon dynamics. If the dynamic parameters are chosen improperly, the system can easily become unstable and unsolvable, in which case no solution exists. Even when the system is solvable, selecting appropriate initial parameters is also critical. In my experiments, I observe cases where poor initialization leads to system explosion, causing the network states to diverge. After redesigning the network architecture, this problem is resolved. The second issue lies in the capital variable k. Our model requires computing terms involving k, so when k approaches zero, the computation becomes numerically unstable. Moreover, the paper does not explicitly state the constraint that $k > 0$. I consider this a critical omission, because it will introduces an inequality constraint on the control variable I. As a result, the Euler equation in [6] should also include the KKT conditions

After sloving 2 questions above, we get $\Theta^* \approx \inf_\Theta \ell$. However, a new question arises: does I^{Θ^*} approximately maximize the lifetime objective? The answer is not necessarily. In the Euler-equation approach, we construct the loss based on the first-order condition (FOC), which is only a necessary condition and does not guarantee global optimality. This is the reason why the lifetime reward obtained via the Euler-equation method is slightly lower than the reward from directly maximizing the lifetime objective (shown in figure 1a).

2.3.2 Control and state with constraints

In real-world , almost all problems involve constraints, so it is essential to take them into account. There is a large body of related research in safe RL that addresses such constraints[11]. As briefly discussed in section 2.3.1, basic model should have a constraint $k_t \geq 0$, i.e., $I_t \geq -(1-\delta)k_t$. Introducing a multiplier $\mu \geq 0$ for this inequality constraint, we get the following Lagrangian. For brevity, I omit the subsequent derivation of the KKT conditions, which can be easily derived from $\mathcal{L}$.

$$\mathcal{L} = \mathbb{E}\Bigg[\sum_{t=0}^{\infty} \beta^t \big(e(k_t, I_t, z_t) - \lambda_t(k_{t+1} - (1-\delta)k_t - I_t\big) - \mu_t(-I_t - (1-\delta)k_t)\Bigg].$$

In stochastic control, imposing constraints on the control variables typically makes the problem significantly more difficult. Fortunately, within deep learning approaches, such constraints can be enforced automatically through the network architecture. However, imposing special structures inevitably makes the DNN harder to train, so designing efficient network architectures becomes an important issue.

2.3.3 Generalization of the model

Research without practice could be empty. How a model performs outside the laboratory is therefore crucial. The final issue to be discussed is the generalization ability of the model. As briefly discussed in section 2.2, k-fold cross validation can be used to test the robustness. As the problem scale increases, we can also examine whether our model obeys a scaling law. If it does, then the model is worth further training.

3 Extension with risky debt

In this section, we denote the stock of debt as b, with a positive value denoting debt, and a negative value denoting cash.

$$\begin{aligned} e(k, k', b, b', z) \; &\equiv (1-\tau)\,\pi(k, z) - \psi(k' - (1-\delta)k,\; k) - \big(k' - (1-\delta)k\big) + \frac{b'}{1+\tilde{r}(z, k', b')} \\ &\quad + \frac{\tau\,\tilde{r}(z, k', b')\,b'}{\big(1+\tilde{r}(z, k', b')\big)(1+r)} - b. \end{aligned}$$

To better understand the problem, we manually derive a control variable D at each time period so that $b' = b + D$. D_t denotes the net debt issuance at time t, that is, the amount of new debt issued minus debt repayments during period. Hence, the reward function becomes

$$e(k, b, z, I, D) \; \equiv (1-\tau)\,\pi(k, z) - \psi(I,\; k) - I + \frac{b'}{1+\tilde{r}(z, k', b')} + \frac{\tau\,\tilde{r}(z, k', b')\,b'}{\big(1+\tilde{r}(z, k', b')\big)(1+r)} - b.$$

where $b' = b + D, k' = (1-\delta)k + I$. Dynamic corporate finance problem can be written as

$$\max_{\{I_t, D_t\}_{t=0}^{\infty}} \mathbb{E}\left[\sum_{t=0}^{\infty} \beta^t e(k_t, b_t, z_t, I_t, D_t)\right]$$

subject to:

$$\begin{aligned} k_{t+1} &= (1-\delta)k_t + I_t, \\ \ln z_{t+1} &= \rho \ln z_t + \epsilon_t, \\ b_{t+1} &= b_t + D_t \end{aligned}$$

where

$$e(k_t, b_t, z_t, I_t, D_t) \equiv (1-\tau)\,\pi(k_t, z_t) - \psi(I_t,\ k_t) - I_t + \frac{b_t + D_t}{1 + \tilde{r}(z_t, (1-\delta)k_t + I_t, b_t + D_t)} + \frac{\tau\,\tilde{r}(z_t, (1-\delta)k_t + I_t, b_t + D_t)(b_t + D_t)}{(1 + \tilde{r}(z_t, (1-\delta)k_t + I_t, b_t + D_t))\beta} - b_t$$

Following a derivation similar to Section 2.1, we obtain the following Lagrangian

$$\mathcal{L} = \mathbb{E}\Bigg[\sum_{t=0}^{\infty} \beta^t \big(e(k_t, b_t, z_t, I_t, D_t) - \lambda_t(k_{t+1} - (1-\delta)k_t - I_t\big) - \gamma_t(b_{t+1} - b_t - D_t)\Bigg].$$

The Euler equation is constructed by the following equations

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial I_t} &= -\psi_I(I_t, k_t) - 1 + \lambda_t = 0, \\
\frac{\partial \mathcal{L}}{\partial k_{t+1}} &= \mathbb{E}\big[\beta(e_k(k_{t+1}, b_{t+1}, z_{t+1}, I_{t+1}, D_{t+1}) + \lambda_{t+1}(1-\delta))\big] - \lambda_t = 0, \\
\frac{\partial \mathcal{L}}{\partial D_t} &= e_D(k_t, b_t, z_t, I_t, D_t) + \gamma_t = 0, \\
\frac{\partial \mathcal{L}}{\partial b_{t+1}} &= \mathbb{E}\big[\beta e_b(k_{t+1}, b_{t+1}, z_{t+1}, I_{t+1}, D_{t+1}) + \gamma_{t+1}\big] - \gamma_t = 0,
\end{aligned}$$

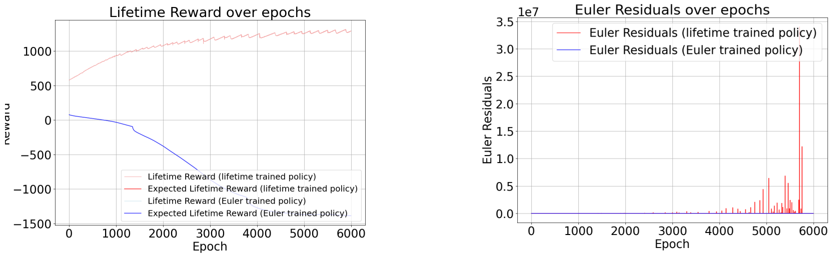
where $t = 0, 1, 2\ldots$, $e_k(k, b, z, I, D) = \partial e(k, b, z, I, D)/\partial k$, $e_D(k, b, z, I, D) = \partial e(k, b, z, I, D)/\partial D$, $e_b(k, b, z, I, D) = \partial e(k, b, z, I, D)/\partial b$. We obtain Euler residual, i.e., loss function, as follows

$$\Xi\Bigg[\big[\beta(e_k(k_{t+1}, b_{t+1}, z_{t+1}, I_{t+1}, D_{t+1}) + \lambda_{t+1}(1-\delta)) - \lambda_t\big]^2 + \big[\beta e_b(k_{t+1}, b_{t+1}, z_{t+1}, I_{t+1}, D_{t+1}) + \gamma_{t+1} - \gamma_t\big]^2\Bigg] \tag{4}$$

where $\lambda_t = \psi_I(I_t, k_t) + 1$, $\gamma_t = -e_D(k_t, b_t, z_t, I_t, D_t)$.

Regarding to risky extension, I have not achieved ideal numerical results, particularly with the Euler residual method. I suspect the potential cause lies in the non-convex optimization landscape; since these algorithms are designed based on FOC, they might be trapped in a saddle point.Introducing risky rate makes the model significantly unstable and more difficult to train. Numerical results are shown in the Figure 2a and 2b. It is worth noting that despite both achieving near-zero Euler residuals (0.927418 and 0.747614), there is a huge difference in their practical performance. This once again underscores the non-convex nature of the optimization landscape.

The concept of using Bellman residuals is very similar to RL. Since RL is a natural fit for dynamic corporate finance problems, I plan to apply DDPG to this model when time permits. I believe it has great potential to yield better results here.



(a) Life-time reward over training epochs

(b) $\sum_{t=0}^{T}(\partial\mathcal{L}/\partial I_t)^2$ over training epochs

Figure 2: Effective metrics

4 Conclusion

In this report, I have formulated the dynamic corporate finance problem as an optimal control framework. This approach not only provides a clearer conceptual understanding but also extends the model's applicability to other potential scenarios. For the basic model, I successfully replicated the three methods from [8] and designed three effective metrics, all of which yielded excellent results.

Subsequently, I identified three key issues. These issues are relevant not only to the basic model but also to a wide range of similar problems, making them significant areas for my future research. Unfortunately, for the model with a risky rate, I encountered a highly complex optimization landscape that remained unresolved. I am convinced that reinforcement learning methods—specifically DDPG and its variants—are well-suited to tackle this challenge, and I plan to explore them in my future work.

Finally, I would like to express my sincere gratitude to the Machine Learning Center of Excellence team at JPMorgan Chase for providing such an intriguing topic, which has been immensely helpful even for my own academic research.

References

[1] J. R. Hicks, *Value and Capital: An Inquiry into Some Fundamental Principles of Economic Theory*. Oxford University Press, 1937.

[2] F. Modigliani and M. H. Miller, "The cost of capital, corporation finance and the theory of investment," *American Economic Review*, vol. 48, no. 3, pp. 261–297, 1958.

[3] R. C. Merton, "On the pricing of corporate debt: The risk structure of interest rates," *Journal of Finance*, vol. 29, no. 2, pp. 449–470, 1974.

[4] H. E. Leland, “Corporate debt value, bond covenants, and optimal capital structure,” *Journal of Finance*, vol. 49, no. 4, pp. 1213–1252, 1994.

[5] H. E. Leland and K. B. Toft, “Optimal capital structure, endogenous bankruptcy, and the term structure of credit spreads,” *Journal of Finance*, vol. 51, no. 3, pp. 987–1019, 1996.

[6] I. A. Strebulaev and T. M. Whited, “Dynamic models and structural estimation in corporate finance,” in *Handbook of the Economics of Finance*, vol. 2A, 2012.

[7] P. Bolton, H. Chen, and N. Wang, “A unified theory of Tobin’s q, corporate investment, financing, and risk management,” *Journal of Finance*, vol. 66, no. 5, pp. 1545–1578, 2011.

[8] L. Maliar, S. Maliar, and P. Winant, “Deep learning for solving dynamic economic models,” *Journal of Monetary Economics*, vol. 123, pp. 155–170, 2021.

[9] H. Wang, X. Zhou. “Continuous-time mean–variance portfolio selection: A reinforcement learning framework”, *Mathematical Finance*, vol. 30, pp. 1273–1308, 2020

[10] M. Dai, Y. Dong, Y. Jia, X.Zhou. “Data-Driven Merton’s Strategies via Policy Randomization” *arXiv preprint*, arXiv:2312.11797 , 2023.

[11] S. Gu, Y. Long , Y. Du, G. Chen, F. Walter, J. Wang, and A. Knoll. “A review of safe reinforcement learning: Methods, theories and applications.” IEEE Transactions on Pattern Analysis and Machine Intelligence (2024).