

Article

A Parallel Monte Carlo Algorithm for the Life Cycle Asset Allocation Problem

Xueying Yang^{1,2}, Chen Li¹ , Xu Li^{1,2} and Zhonghua Lu^{1,*} 

¹ Computer Network Information Center, Chinese Academy of Sciences, Beijing 100083, China; yangxueying@cnic.cn (X.Y.); lichen@sccas.cn (C.L.); lixu@cnic.cn (X.L.)

² University of Chinese Academy of Sciences, Beijing 100190, China

* Correspondence: zhlu@cnic.cn

Abstract: Life cycle asset allocation is a crucial aspect of financial planning, especially for pension funds. Traditional methods often face challenges in computational efficiency and applicability to different market conditions. This study aimed to innovatively transplant an algorithm from reinforcement learning that enhances the efficiency and accuracy of life cycle asset allocation. We synergized tabular methods with Monte Carlo simulations to solve the pension problem. This algorithm was designed to correspond states in reinforcement learning to key variables in the pension model: wealth, labor income, consumption level, and proportion of risky assets. Additionally, we used cleaned and modeled survey data from Chinese consumers to validate the model's optimal decision-making in the Chinese market. Furthermore, we optimized the algorithm using parallel computing to significantly reduce computation time. The proposed algorithm demonstrated superior efficiency compared to the traditional value iteration method. Serial execution of our algorithm took 29.88 min, while parallel execution reduced this to 1.42 min, compared to the 41.15 min required by the value iteration method. These innovations suggest significant potential for improving pension fund management strategies, particularly in the context of the Chinese market.

Keywords: life cycle asset allocation; Monte Carlo; glide path; Bellman equation; parallel optimization



Citation: Yang, X.; Li, C.; Li, X.; Lu, Z. A Parallel Monte Carlo Algorithm for the Life Cycle Asset Allocation Problem. *Appl. Sci.* **2024**, *14*, 10372. <https://doi.org/10.3390/app142210372>

Academic Editor: Asterios Bakolas

Received: 2 May 2024

Revised: 12 October 2024

Accepted: 16 October 2024

Published: 11 November 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In the expansive field of computational finance, the topic of life cycle asset allocation stands out as particularly significant. This is especially true given its alignment with the evolving third pillar of China's pension system—a current area of vulnerability. In the context of China's pension system, the application of life cycle asset allocation is particularly crucial. It not only represents an innovation in traditional pension management models but also serves as a vital approach to addressing current system challenges and enhancing the level of retirement security. The Chinese pension system is primarily composed of three pillars: the first pillar is the Basic Pension Insurance, which has broad coverage but limited benefit levels; the second pillar includes Enterprise Annuities and Occupational Annuities, which, despite achieving certain results, have relatively narrow coverage and face issues such as a limited fund scale and imbalanced development structure; the third pillar consists of supplementary pension insurance represented by personal pensions, which is still in its early stages with enormous potential.

However, this system faces numerous challenges. Firstly, life insurance products are highly homogenized, failing to meet the diverse needs of retirement. Many products lack innovation in their design, with underlying assets heavily reliant on fixed-income products such as bonds and time deposits. In the current context of global economic slowdown and continuously declining interest rates, the sustainability of these products is under severe threat. Low yields are insufficient to effectively counteract inflation, which, over time, will erode the real purchasing power of pensions and affect the quality of life of the elderly. Therefore, to address this shortcoming and manage potential future market

risks, the development of personalized pension products that balance returns and risks is urgently needed. Life cycle asset allocation is precisely such a strategy. Optimal asset allocation strategies involve the glide path concept [1], which refers to calculating the changing annual investment proportion in risky assets based on the specific life cycle and financial market of the investor throughout an investor's life cycle. The optimal problem is typically modeled as a stochastic process, examined through a set of inter-connected random variables, such as the Markov process. The development of algorithms calculating for glide path and efficient problem-solving for these models is increasingly emphasized due to the complex nature of life cycle asset allocation.

Historically, the foundational work in classical life cycle fund models was initiated with Markowitz's mean-variance model [2], which pioneered the integration of mathematical statistics into effective asset allocation within modern investment theories. This was extended by Modigliani [3], who proposed the life cycle hypothesis, suggesting that investors should strategically manage their consumption and investment throughout their lifespan with a view toward future earnings. Building on this, numerous academics have further developed life cycle investment mechanisms, providing effective investment guidance and ensuring retirement security. A significant advancement in life cycle fund development was achieved by Merton [4], who advocated the use of utility functions to capture the investment and consumption preferences of individual investors. This was enhanced by Bodie et al. [5], who emphasized the critical relationship between an investor's income and their investment decisions by incorporating human capital considerations. This perspective suggests that younger investors, with greater career flexibility, are likely to have a higher tolerance for risk. Moreover, various models based on life cycle investment strategies, such as those by Cocco et al. [6] and Campbell et al. [7], have gained significant recognition. Gomes, F. et al. [8,9] introduced borrowing constraints and bequest motives. These models account for external factors like labor income risk, which can affect investors. However, some of these assumptions for the complexity of human financial behavior and stochastic market conditions do not appear to reflect empirical observations.

Blake et al. [10] apply the assumption of the Epstein–Zin utility function. Firstly, the analytical resolution of such intricate problems, often distilled into a Bellman equation framework, encounters insurmountable hurdles due to the intricacies of expectation terms and the lack of closed-form solutions. The Bellman equation, central to stochastic control and decision-making, faces challenges such as mathematical complexity and computational inefficiency, while also presenting significant opportunities for advancement in various fields [11]. The equation's application in infinite-dimensional spaces and under conditions of uncertainty introduces unique mathematical difficulties and necessitates innovative approaches to ensure solution uniqueness and optimality [12,13]. Furthermore, Gerstenberg, Neininger et al. [14] have expanded the scope of Bellman equations into the realm of distributional reinforcement learning (RL). In macroeconomic utility maximization problems, Shigeta emphasizes the Bellman equation's role in modeling consumer behavior under uncertainty and constraints [15]. Fei et al. [16–18] proposed an Exponential Bellman Equation for improved regret bounds, while Jones and Peet [19] generalized Bellman's equation for path planning and obstacle avoidance. Beck et al. [20,21] introduced efficient nonlinear Monte Carlo methods to tackle the computational challenges in high-dimensional stochastic optimal control. Becker et al. [22–24] address the challenge of solving high-dimensional optimal stopping problems using deep learning techniques. Kristensen et al. [25] address the verification of a continuous-time utility maximization problem, which is a common framework in macroeconomics. Despite the effectiveness of various algorithms in dynamic programming and reinforcement learning, such as value iteration, these models exhibit significant limitations in addressing high-dimensional state spaces and complex stochastic environments inherent in pension planning. Traditional models, like value iteration and policy iteration, suffer from high computational complexity and slow convergence, especially in high-dimensional state spaces. To address these issues, we propose a Monte Carlo method that effectively manages high-dimensional state and decision spaces, offers

flexibility with different stochastic models, and ensures faster convergence for solving specific challenges. This approach provides a robust solution to the computational and modeling deficiencies of existing methods, ensuring efficient and accurate pension planning. Secondly, the models are based on American investors, lacking examination of the effectiveness on Chinese investors. While advancements in Monte Carlo planning and simulation by Silver et al. [26], Fatica et al. [27], and Abbas-Turki et al. [28] provide robust methods for navigating the probabilistic nature of asset allocation.

With the advancement of reinforcement learning, we innovatively correspond the concepts of states and actions in reinforcement learning with the proportions of risky assets and investment behaviors in life cycle asset allocation, matching optimization strategies with glide paths, introducing a Monte Carlo-based algorithm to solve the Markov process represented as a Bellman equation. It facilitates the exploration of state and policy spaces through random sampling, thus enabling the evaluation of the expected returns of various strategies across a multitude of possible scenarios. Addressing robustness, model complexity, and computational demands is crucial, necessitating advanced algorithms where high-performance computing plays a vital role. Contributions by Mao et al. [29] and Weng et al. [30] in improving computational frameworks through Poly-hoot and Envpool, respectively, demonstrate the critical role of high-performance and parallel computing in enhancing model performance and efficiency. Incorporating these advancements into the life cycle asset allocation discourse, this paper parallelizes the proposed algorithm to overcome computational bottlenecks and meet application requirements. Furthermore, we build a labor income model for Chinese investors, aiming to bridge the gap between theoretical models and their practical implementation in the fast-evolving landscape of computational finance. Our innovative approach demonstrates substantial advancements in life cycle asset allocation by (1) transplanting reinforcement learning algorithms to the pension problem, (2) utilizing Chinese consumer data for validation, and (3) optimizing the algorithm through parallel computing.

The rest of this paper is organized as follows: Initially, we will succinctly outline the life cycle asset allocation model being examined. Following this, we will elaborate on our Monte Carlo-centric algorithm employed to adeptly navigate the Bellman equation's challenges. Subsequently, we present experimental results, the parallel methods and their effectiveness, along with detailed analyses. Finally, this paper will conclude with an overview and suggestions for future research directions.

2. Materials and Methods

2.1. Life Cycle Asset Allocation Model

2.1.1. Utility Function Selection

We adopt the utility function and financial asset used in Blake et al. [8]. Consumer preferences are in Epstein–Zin preferences, the utility function, recursive form:

$$U_t = \{(1 - \beta)C_t^{1-1/\varphi} + \beta p_t (E_t[U_{t+1}^{1-\gamma}])^{(1-1/\varphi)/(1-\gamma)}\}^{1/(1-1/\varphi)} \quad (1)$$

In this context, U_t is the level of utility at age t ; C_t is the level of consumption at age t ; p_t is the (non-random) one-year survival probability at age t , i.e., the probability that a member alive at age t will live to age $t + 1$; γ is the relative risk aversion coefficient (RRA); φ is the intertemporal elasticity of substitution (EIS); and β is an individual's one-year personal discount rate.

Given the risk of death at age t , the member is assumed to have a maximum potential age of T years. Thus, in the final year, $p_T = 0$. The termination condition of the utility function is as follows:

$$U_T = \{(1 - \beta)C_T^{1-1/\varphi}\}^{1/(1-1/\varphi)} \quad (2)$$

Epstein–Zin preferences are able to separate relative risk aversion (RRA) from intertemporal elasticity of substitution (EIS), where individuals with high risk aversion want to avoid consumption uncertainty in a given period, this being the avoidance of consump-

tion reductions required in unfavorable states of nature (e.g., a large drop in stock prices). Individuals with low EIS want to avoid consumption fluctuations over time, in particular avoiding consumption reductions compared to the previous time period to a decrease in consumption.

2.1.2. Financial Asset

Two types of financial assets are assumed, a risky equity fund and a risk-free bond fund. By choosing to invest in the bond fund, which has a constant annual real return \bar{R}_f , and the equity fund, which has a return from age t to $t + 1$, a portion of the pension fund α_t is invested in the risky asset at age t , the total return R_t is shown as follows:

$$R_t = \bar{R}_f + \alpha_t(\mu + \sigma Z_{1,t}) \tag{3}$$

where $0 \leq \alpha_t \leq 1$ since the risky assets shall not be sold out; μ is the annual risk premium of the risky asset (a deterministic parameter); σ is the annual volatility of the return of the risky asset (a deterministic parameter); $\{Z_{1,t}\}$ is a series of independent identically distributed standard normal random variables. Prior to retirement, the pension fund's total pension wealth W_{t+1} has the following recurrence relationship:

$$W_{t+1} = (1 + R_t - c_t) \times (W_t + Y_t - C_t) \geq 0, \text{ start} \leq t \leq \text{retirement} \tag{4}$$

$$W_{t+1} = (1 - \alpha_t) \times W_t / \ddot{a}_t \times \ddot{a}_{t+1} + [\alpha_t W_t + (1 - \alpha_t) \times W_t / \ddot{a}_t - C_t] \times (1 + \bar{R}_f + \mu + \sigma Z_{1,t}), \tag{5}$$

$\text{start} \leq t \leq \text{retirement}$

The wealth of the pension shall never be negative. Labor income Y_t will be discussed in Section 2.1.3. c_t is the custodian fee, the pension target fund fee structure is as follows:

Self-charges:

Subscription/subscription fees (a one-time charge is not repeated, and the subscription fee rate is 1.2% on average), management fees, and custodian fees (according to the Wind system, as of the end of 2019, the weighted average of the management fee rate is 0.81%; the weighted average of the custodian fee rate is 0.17%, and the two fees add up to a rate of 0.98%).

Underlying fund fees:

Subscription/subscription fees (generally between 0.8% and 1.5%, and large FOF products have the advantage of preferential purchase fees), management fees, custodian fees (according to the Wind System, the weighted average of the management fee rate for equity-type open-end funds was 0.76%; the weighted average of the management fee rate for bond and hybrid open-end funds was 0.33%).

2.1.3. Labor Income Patterns in China: Insights from the China General Social Survey (CGSS)

Labor income is usually an important component of total assets. The growth rate of labor income from age t to $t + 1$ is given by g_t . Two aspects of labor income present risk—the systematic volatility of labor income and the correlation between labor income growth and stock returns. Labor income Y_{t+1} earned at age $t + 1$ is shown below [31]:

$$Y_{t+1} = Y_t \times \exp((S_{t+1} - S_t) / S_t) \tag{6}$$

where $Y_{start} = 1$; S_t is the labor income.

To analyze current trends and patterns within the context of labor income dynamics, we use robust statistical methods to ensure accurate and insightful results. The China General Social Survey (CGSS) is one of the most comprehensive, continuous national social surveys conducted in China. In our study, we utilize the most recent dataset, specifically the 2021 data, to model labor income. The 2021 dataset provides a contemporary snapshot of the economic conditions influencing labor income, making it a vital resource for understanding the factors that impact earnings across different demographics and regions. We process

the data by the age and the income columns, dropping the unavailable and the extreme sample points and computing the average labor income per year for investors in China. We use a degree 3 polynomial function to estimate labor income, $S_t = -1745.40 + 141.43 \times t - 3.13 \times t^2 + 0.021 \times t^3$, shown in Figure 1.

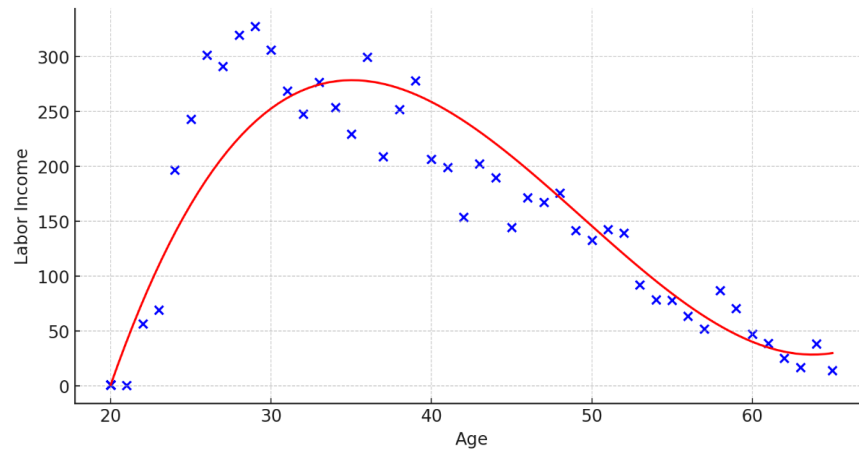


Figure 1. Labor income estimation.

Many studies have hypothesized that real income follows a hump-like pattern, characterized by rapid growth until around age 35, a slower rise between ages 45 and 50, and a gradual decline thereafter. This pattern is widely accepted as a general model of income dynamics over the life cycle. However, through our detailed data cleaning and modeling of the China General Social Survey (CGSS), we have uncovered unique characteristics in the labor income trajectory of Chinese consumers, as depicted in the figure. The graph illustrates labor income against age, highlighting several key deviations from the expected pattern. Initially, labor income increases rapidly, reaching its peak earlier than the global average, around age 30–35. This peak is followed by a relatively sharp decline, contradicting the hypothesis of a slower rise until the age of 50. Instead, the data show a continuous decrease in labor income after the early thirties, which becomes more pronounced beyond the age of 40. By age 60, labor income levels are significantly lower, indicating a steep downward trend.

This distinctive pattern can be attributed to several socio-economic factors unique to China, including retirement policies, the structure of the labor market, and cultural attitudes toward work and retirement. The earlier peak and subsequent decline suggest that Chinese workers may experience a more pronounced reduction in income as they age compared to their counterparts in other countries. Figure 1 visually presents these findings, emphasizing the rapid increase in labor income up to the early thirties, the peak around age 30–35, and the subsequent decline, highlighting the unique labor income trajectory of Chinese consumers. By examining these data, we can gain a deeper understanding of the economic behavior and challenges faced by Chinese workers throughout their careers.

2.1.4. Optimization Problem

The model has two control variables at each age t , the proportion of risky asset allocation α_t , and the level of consumption C_t . The objective function is $\max_{\alpha_t, C_t} U_t$.

(1) Before maturity of the product:

$$0 \leq \alpha_t \leq \alpha_{\max} \tag{7}$$

When $start \leq t \leq retirement$, $C_t \leq Y_t$. Each year, individuals are not allowed to borrow money from the fund, therefore the consumption shall be less than the labor income. And the contribution rate shall be larger than or equal to zero.

When $retirement \leq t \leq T$, $C_t \leq (1 - \alpha_t) \times W_t / \bar{a}_t + \alpha_t W_t$;

(2) Obtain the Bellman equation:

$$V_t = \max_{\alpha_t, C_t} \{ (1 - \beta) C_t^{1-1/\varphi} + \beta p_t (E_t [V_{t+1}^{1-\gamma}])^{(1-1/\varphi)/(1-\gamma)} \}^{1/(1-1/\varphi)} \tag{8}$$

Since no analytic solution exists, a numerical solution method must be used to derive the value function and the corresponding optimal control parameters [13]. The utility function termination condition is used to compute the corresponding value function for the previous cycle, and the process is iterated backward following a standard dynamic programming strategy.

2.2. Description of Monte Carlo-Based Algorithm for Solving the Bellman Equation

We first come up with this Monte Carlo-based algorithm to solve this pension problem, making use of the idea of reinforcement learning and making it possible and applicable to make the solving process to parallelization. We consider the proportion of risky asset α_t as the rows, and therefore we discrete it as 20 intervals. In addition, we assume the consumption rate C_t as the columns, and discrete it as 20 intervals. The state s at age t is (α_t, C_t) . The action a represents the movement (right, left, up, and down). To make the process more robust, we make the movement continuously with one interval a time, the next state at age $t + 1$ is (α_{t+1}, C_{t+1}) . Each reward, where the utility U_t comes from, is the function of C_t .

When a state is encountered, choosing a certain behavior with a certain probability to reach the next state and obtaining the corresponding reward is a randomness strategy. There must be at least one optimal strategy among all the strategies. The sum of the multiple subsequent rewards is the payoff. It is not appropriate enough to measure the goodness of a strategy in terms of rewards.

Specifying a state, taking a stochastic strategy, and then weighting the rewards by an average—which is the expectation—gives the state value function. In addition to this, there is another type of value function, which is the state action value function.

Our solution algorithm is formulated in Algorithm 1:

Algorithm 1: Monte Carlo Method for Solving the Bellman Equation

```

1 : Initialization  $Q(s = (\alpha, C), a) \leftarrow 0, \forall s \in S, a \in A$ 
2 : Set  $N(s = (\alpha, C), a) \leftarrow 0, \forall s \in S, a \in A$ 
3 : Set  $\epsilon \leftarrow 1, k \leftarrow 1$ 
4 :  $\pi_k \leftarrow \epsilon$ -greedy( $Q$ )
5 : For  $k < 1000$  do
6 : Generate episode  $E_k = (s_1, a_1, r_2, \dots, s_T)$  following  $\pi_k$ , while for
   start  $\leq t \leq retirement$   $C_t < Y_t$ , for  $retirement \leq t \leq T$ ,  $C_t \leq (1 - \alpha_t) \times W_t / \bar{a}_t + \alpha_t W_t$ 
    $0 \leq \alpha_t \leq \alpha_{max}$ .
7 : for each  $(s_t, a_t) \in E_k$  do
8 :      $N(s_t, a_t) \leftarrow N(s_t, a_t) + 1$ 
9 :      $G_t = r_{t+1} + \gamma r_{t+2} + \dots + \gamma^k r_{t+1+k}$ 
10 :     $Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \frac{1}{N(s_t, a_t)} (G_t - Q(s_t, a_t))$ 
11 : end for
12 :  $k \leftarrow k + 1$ 
13 :  $\epsilon \leftarrow \frac{1}{k}$ 
14 :  $\pi_k \leftarrow \epsilon$ -greedy( $Q$ )
15 : end for

```

To address high-dimensional and iterative problems, we used several methods. State space discretization was our first approach. We divided the state space (pension wealth) into a finite number of points, transforming a continuous problem into a discrete one. This

reduces computational complexity. The Monte Carlo method simulated many future state paths. It estimates expectations in high-dimensional spaces.

Dynamic programming was also applied, which iterates backward from the end-point to the starting point. This progressively solves the value function, ensuring the optimal strategy at each step. We used interpolation methods, like linear interpolation, to smooth transitions between state points. This maintains accuracy in the value function calculations. Consistent results were ensured by setting the same random seed for Monte Carlo simulations.

In setting the state space, we avoided computational issues by starting from 0.01 instead of 0. This prevents problems like division by zero. The range ends at 1000, making calculations and interpretations easier. We generated 100 state points to capture wealth changes accurately. This balance ensures sufficient granularity without overwhelming computational resources.

We chose pension wealth for the state space because it influences consumption and investment decisions, which determines how much members can consume and invest. Pension wealth also affects the trade-offs between risk and return. These levels change over time due to investment returns, consumption, and labor income. Capturing these dynamics is essential for optimizing long-term decisions. In dynamic programming, the pension wealth is the state variable. It describes the current financial status. The state transition equation dictates the changes over time, guiding optimal strategy calculations.

The utility function is central to dynamic programming and the Bellman equation. It measures satisfaction in each state. In our pension problem, it reflects preferences for consumption and risk aversion. The utility function quantifies immediate utility and expected future utility. Immediate utility indicates satisfaction from current consumption. Expected utility combines immediate utility with the future value function. Monte Carlo simulations help calculate this. Interpolating the future state's value function aggregates immediate utility and the weighted future value. This provides a comprehensive measure of overall utility.

Managing the exponential expectation terms in Equation (8) required special handling. Avoiding zero values was crucial, as zero's negative exponentiation can lead to undefined results. To explore the implications of different consumption paths on the overall utility, we set the initial value function for the final year as the terminal value of the utility function. This approach ensures that the terminal value reflects the cumulative effect of all prior consumption decisions. Initialization: set the initial value function at T using the terminal utility function U_T . Random Generation of C : for period T , generate C_T randomly from a chosen probability distribution.

This Monte Carlo-based algorithm for solving the Bellman equation operates on the principle of learning from complete episodes of interaction with an environment. At each state within an episode, a decision is made to take an action, guided by the probability ϵ , leading to a subsequent state and an associated reward. This sequence continues until a terminal state is reached, capturing a full trajectory of states, actions, and rewards. The goal is not to evaluate the quality of a strategy based solely on immediate rewards but to consider the long-term payoffs, which are captured in the state-action value function Q . This function estimates the expected return of taking an action in a given state and following the current policy thereafter. By averaging the returns over multiple samples, the algorithm approximates the expected value, thus providing an estimate of the state-action value function that underlies optimal decision-making according to the Bellman equation. This stochastic approach, incorporating randomness in the strategy choice, is critical for exploring the action space to find an optimal strategy among all possible ones.

3. Results

3.1. Glide Path

The glide path depicted in the following graph illustrates the strategic asset allocation for a pension target-date fund, emphasizing a decreasing risk profile as participants age. The parameters we use for the model are shown in Table 1.

Table 1. Model Parameters.

Parameter	Value
μ	0.04
σ	0.2
γ	5.0
φ	0.2
β	0.96
<i>start</i>	20
<i>retirement</i>	65
T	120
p_t	0.99

Figure 2 shows the calculated glide path curve. This risk mitigation strategy is commonly employed in life cycle or target-date funds, where the equity ratio is highest during the early working years when the investor’s age is below 50. This allows for a more aggressive investment stance, capitalizing on the potential for higher returns that equities can offer over the long term, while the investor can endure short-term volatility.

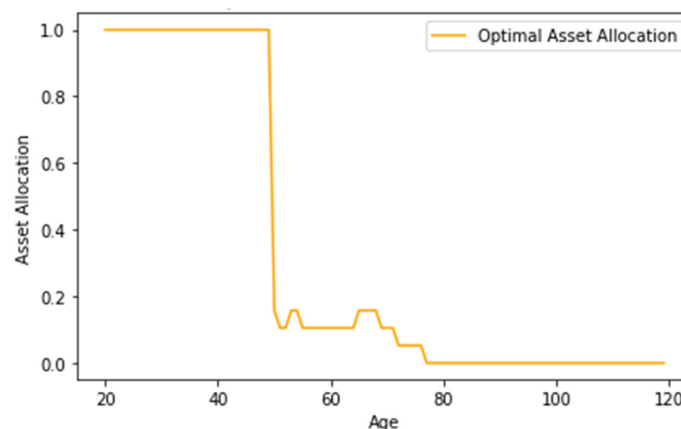


Figure 2. Glide path.

As the individual nears retirement, beginning approximately at age 55, the optimal equity allocation rate begins to decrease steadily. This gradual transition, known as the “glide path”, is designed to systematically reduce the fund’s risk exposure as the investor’s capacity to recover from market downturns diminishes with shorter investment horizons. By the time the investor reaches age 75, the fund has shifted most of its holdings into more conservative investments, such as bonds and cash, to preserve capital and provide income stability for the forthcoming retirement years. The graph effectively encapsulates this prudent approach to balancing growth and security through age-adjusted asset allocation.

Figure 3 illustrates the trend of optimal consumption over time. The graph shows a consistent upward trajectory during the initial periods, reflecting a period of increasing consumption. Eventually, the consumption level stabilizes, indicating an equilibrium or a steady state.

Figure 4 presents the evolution of pension wealth over time comparing it to the 60/40 strategy. Initially, there is a steady accumulation of pension wealth, represented by a significant upward trend, suggesting contributions or growth in value. This rise peaks

before a sharp decline is observed, eventually reaching a point where pension wealth drops to a stable, near-zero level. This behavior highlights the life cycle of pension wealth accumulation and decumulation, reflecting typical retirement consumption patterns.

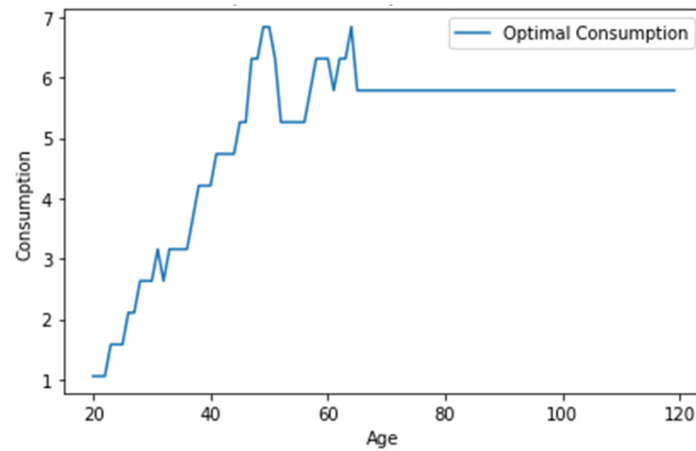


Figure 3. Optimal consumption.

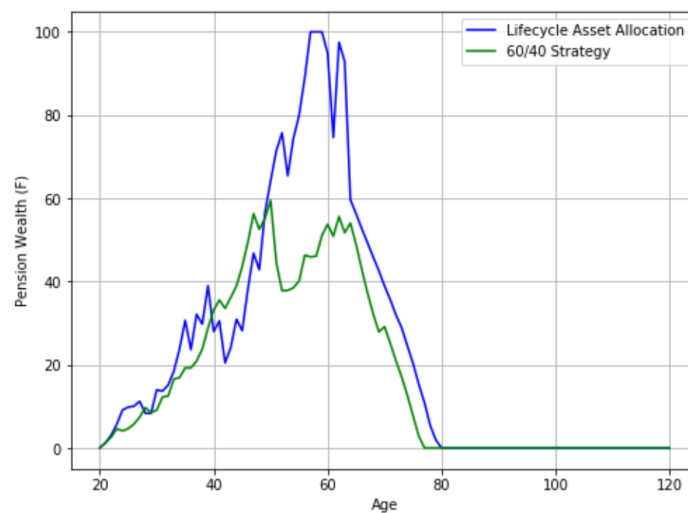


Figure 4. Pension wealth.

3.2. Parallel Monte Carlo

In this study, we implemented a parallel Monte Carlo simulation to solve a Bellman equation, which is essential for decision-making under uncertainty in finance. We utilized the mpi4py library to leverage the distributed computing capabilities of the high-performance computing (HPC) environment. Specifically, our experiments were conducted on a cluster with 256 AMD EPYC 7773X 64-Core Processors, Table 2 shows the server parameters used.

Table 2. Server Parameters.

Server	Parameters
Nodes	AMD EPYC 7773X 64-Core Processor
Operating	Centos7.6
MPI	hpcx-2.4.1
Network	HDR Infiniband (200 Gb)

To enhance the efficiency of our parallel algorithm, we adopted a hybrid memory approach that exploits both distributed and shared memory systems. The hybrid approach

allowed us to minimize the communication overhead between processes by effectively managing the data locality, thus reducing the time spent in data exchange and synchronization across the processors.

Additionally, we optimized memory access patterns by aligning the data structures with the cache line, leading to a reduction in cache misses and improved data throughput. By carefully orchestrating the computational load and communication among the processors, we achieved a significant reduction in the overall runtime, demonstrating the scalability of our simulation across multiple nodes.

The optimization measures were specifically tailored to the computational characteristics of Monte Carlo methods, where the primary focus was on load balancing and reducing the random memory access patterns that could lead to bottlenecks in a high-performance computing setup. The randomness inherent in the Monte Carlo simulation often results in non-uniform memory access patterns. To address this, we implemented an intelligent scheduling system that dynamically adjusts the allocation of tasks based on the current load of each processor, thus ensuring that no single processor becomes a point of contention, leading to a more efficient parallel execution.

Our implementation demonstrates the utility of parallel optimization techniques such as reducing communication costs through intelligent data distribution, optimizing memory access, and implementing dynamic load balancing. These optimizations are crucial for simulations that require a substantial number of iterations to obtain high-precision results, as is the case with the evaluation of the expected utility in asset allocation strategies. Table 3 shows a time comparison between the value iteration method and our proposed method after parallelization versus serialization. The value iteration method shares the same parameters as the model.

Table 3. Computing time.

	Thread Number	Computing Time (min)
Value Iteration Method	1	41.15
	1	29.88
Our Method (Proposed)	2	15.74
	4	8.06
	8	4.17
	16	2.69
	32	1.42

The value iteration method was conducted on a discretized grid of state variable, pension wealth, and it employed a backward induction approach to solve for value functions and policies. The Gauss–Hermite quadrature method with nine nodes was used to capture the stochastic shock dynamics influencing equity returns and income growth. To further refine the accuracy, the variable space was represented by 30 grid points for pension wealth, 10 grid points for labor income, and 20 points each for consumption and asset allocation.

Initialization Stage: At age 120, we define the terminal value function $V[T]$. At this terminal age, the initial terminal utility U_T is calculated based on the given pension wealth and possible consumption levels. The maximum possible consumption C is equal to the total pension wealth W , and this value is chosen to initialize the terminal utility.

Dynamic Programming and Backward Induction: We iterate backward from age 119 to age 20 to compute the optimal value function and corresponding decisions at each age. For each age, we initialize a matrix $V[age]$ to store the current value function and set up policy matrices $policyC[age]$ and $policy\alpha[age]$ to record the optimal consumption and asset allocation proportions, respectively.

Computation of the Value Function: For each state variable (pension wealth W), at each time point, different combinations of consumption C and asset allocation proportion a are explored. Through a nested loop, all possible combinations of consumption and asset allocation are evaluated, and the utility U for each combination is computed. At each

iteration, the immediate utility U from consumption is calculated, along with the expected future value expected V .

Immediate utility U is defined by a power function of consumption C , factoring in a time preference parameter β . The expected future value expected V is calculated using Gauss–Hermite quadrature, which integrates over possible future outcomes by simulating changes in assets based on the distribution of returns.

Finding the Optimal Decisions: For each combination of consumption and asset allocation, the total utility is computed. After iterating through all combinations, the optimal consumption and asset allocation that correspond to the maximum utility are stored in the policy matrices, while the maximum utility is recorded in $V[age]$.

Iteration and Recording of Optimal a and C Strategies:

Through the backward iteration, we ultimately derive the optimal strategies $policyC[age]$ and $policy\alpha[age]$ for each age. These strategies represent the optimal consumption level and asset allocation proportion at every age, determined by selecting the combinations of C and a that maximize utility.

Simulation of the Optimal Path:

Once the optimal strategies for each age are determined, we can simulate the optimal paths of wealth, consumption, and asset allocation over the life cycle. Starting with initial pension wealth and salary income, the optimal consumption C and asset allocation proportion α at each age are derived from the previously calculated policies $policyC[age]$ and $policy\alpha[age]$. Simultaneously, the pension wealth W and labor income Y are updated iteratively using shocks Z_1 and Z_2 , recording the changes over time.

The effectiveness of the parallel optimization techniques employed in our Monte Carlo simulation is evidenced by the marked reduction in computing times with the increasing numbers of threads. When executed on a single thread, the simulation took approximately 29.88 min to complete. Through the implementation of parallel processing, this time was more than halved to 15.74 min with two threads. The trend of decreasing computational time continued as more threads were introduced, with four threads cutting the time down to 8.06 min, an almost four-fold improvement over the single-threaded execution. With 32 threads, the time was further reduced to 1.42 min. This represents a remarkable twenty-fold decrease from the initial time, highlighting the scalability of our parallel algorithm. These times reflect a near-linear speed-up with increasing thread counts, suggesting efficient utilization of computational resources and minimal overhead from parallelization. Our results validate the implementation of our parallel optimization strategies, making a strong case for their effectiveness in high-performance computing contexts, particularly in complex financial simulations where time efficiency is paramount.

4. Discussion

Life cycle asset allocation dynamically adjusts the proportion of various assets (including stocks, bonds, etc.) in the investment portfolio based on factors such as an individual's age, financial status, and market conditions, aiming to achieve the optimal balance between risk and return. Our research indicates that asset allocation based on Chinese consumer data similarly follows a downward-sloping curve. During younger years, when the capacity for risk is higher, the proportion of high-risk assets such as stocks can be moderately increased to seek higher growth in returns. This finding aligns with consumer data in Western countries. As age increases and retirement approaches, the proportion of risk assets should be gradually reduced, and the allocation of more stable, principal-protected assets should be increased to ensure the safety and liquidity of the pension. We observe that consumption behavior aligns closely with the labor income trajectory discussed above. During the early years (up to age 30), as labor income rapidly increases, optimal consumption also rises steadily. This period reflects a phase where individuals prioritize current consumption over long-term savings due to the rapidly increasing income. The decline in labor income becomes more pronounced beyond age 40 but the consumption level remains relatively steady, suggesting effective financial planning and saving strategies that allow individuals

to maintain their consumption levels despite decreasing income. As individuals approach retirement age (around 60), there is a marked drop in consumption, which aligns with the significant reduction in labor income. This drop reflects the transition from reliance on labor income to dependence on accumulated savings and pension funds. After retirement, consumption is maintained through pension income, which helps to smooth consumption and support individuals in their retirement years.

The \ddot{a}_t is computed by the PMA92 Table, which is a mortality table for male pension annuitants in the UK and may not be suitable for Chinese investors. In the future, we also could include other investment targets such as REITs.

5. Conclusions

This paper has made significant advancements in the field of life cycle asset allocation. We innovatively transplanted an algorithm from reinforcement learning, which combines tabular methods and Monte Carlo simulations, to solve the pension problem. Our approach markedly reduces computation time compared to the value iteration method. Additionally, we validated it with data from China, offering detailed insights into risk and returns crucial for real-world asset allocation strategies in the Chinese market. Our experiments derived a customized life cycle asset allocation for Chinese investors aimed at maximizing optimal utility, thereby preserving and appreciating pension assets, providing a more robust economic guarantee for the retirement life of the elderly. Further, by leveraging high-performance computing, we tackled the complex stochastic nature of financial markets by accelerating the process, showcasing the algorithm's efficiency. However, the model requires substantial computing resources. Future work should focus on improving the specific Monte Carlo algorithm with state-of-the-art reinforcement learning methods to make the asset allocation problem more efficient and accurate.

Author Contributions: Conceptualization, C.L.; Methodology, X.Y.; Formal analysis, X.L.; Writing—review & editing, Z.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the Beijing Natural Science Foundation (Grant No. 4232039).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Restrictions apply to the availability of these data. Data were obtained from National Survey Research Center and are available <http://cgss.ruc.edu.cn/> with the permission of National Survey Research Center.

Acknowledgments: This work was supported by the Beijing Natural Science Foundation (Grant No. 4232039). The numerical calculations in this study were carried out on the ORISE Supercomputer.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Yang, X.; Li, C.; Chen, Y.; Lu, Z. A Survey of Models and Algorithms of Numerical Methods Based on Pension Target Funds. *Front. Data Comput.* **2023**, *5*, 85–96. (In Chinese)
2. Markowitz, H. Portfolio selection. *J. Financ.* **1952**, *7*, 77–91.
3. Modigliani, F. The life cycle hypothesis of saving, the demand for wealth and the supply of capital. *Soc. Res.* **1966**, *33*, 160–217.
4. Merton, R.C. Lifetime portfolio selection under uncertainty: The continuous-time case. *Rev. Econ. Stat.* **1969**, *51*, 247–257. [[CrossRef](#)]
5. Bodie, Z.; Merton, R.C.; Samuelson, W.F. Labor supply flexibility and portfolio choice in a life cycle model. *J. Econ. Dyn. Control* **1992**, *16*, 427–449. [[CrossRef](#)]
6. Cocco, J.F.; Gomes, F.J.; Maenhout, P.J. Consumption and portfolio choice over the life cycle. *Rev. Financ. Stud.* **2005**, *18*, 491–533. [[CrossRef](#)]
7. Campbell, J.Y.; Feldstein, M. *Risk Aspects of Investment-Based Social Security Reform*; University of Chicago Press: Chicago, IL, USA, 2000.
8. Gomes, F.; Michaelides, A. Life-cycle asset allocation: A model with borrowing constraints, uninsurable labor income risk and stock-market participation costs. *Uninsurable Labor Income Risk Stock. Mark. Particip. Costs* **2002**. [[CrossRef](#)]

9. Gomes, F.; Michaelides, A. Optimal life-cycle asset allocation: Understanding the empirical evidence. *J. Financ.* **2005**, *60*, 869–904. [[CrossRef](#)]
10. Blake, D.; Wright, D.; Zhang, Y. Age-dependent investing: Optimal funding and investment strategies in defined contribution pension plans when members are rational life cycle financial planners. *J. Econ. Dyn. Control* **2014**, *38*, 105–124. [[CrossRef](#)]
11. Zhao, Z. Variants of Bellman equation on reinforcement learning problems. In Proceedings of the 2nd International Conference on Artificial Intelligence, Automation, and High-Performance Computing (AIAHPC 2022), Zhuhai, China, 25–27 February 2022; Volume 12348, pp. 104–111.
12. Rantzer, A. Explicit solution to Bellman equation for positive systems with linear cost. In Proceedings of the 2022 IEEE 61st Conference on Decision and Control (CDC), Cancun, Mexico, 6–9 December 2022; pp. 2678–2683.
13. Cosso, A.; Gozzi, F.; Kharroubi, I.; Pham, H.; Rosestolato, M. Master Bellman equation in the Wasserstein space: Uniqueness of viscosity solutions. *Trans. Am. Math. Soc.* **2024**, *377*, 31–83. [[CrossRef](#)]
14. Gerstenberg, J.; Neininger, R.; Spiegel, D. On solutions of the distributional Bellman equation. *arXiv* **2022**, arXiv:2202.00081. [[CrossRef](#)]
15. Shigeta, Y. A continuous-time utility maximization problem with borrowing constraints in macroeconomic heterogeneous agent models: A case of regular controls under Markov chain uncertainty. *Available SSRN* **2023**, 4510320. [[CrossRef](#)]
16. Fei, Y.; Yang, Z.; Chen, Y.; Wang, Z. Exponential Bellman equation and improved regret bounds for risk-sensitive reinforcement learning. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 20436–20446.
17. Fei, Y.; Yang, Z.; Chen, Y.; Wang, Z.; Xie, Q. Risk-sensitive reinforcement learning: Near-optimal risk-sample tradeoff in regret. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 22384–22395.
18. Fei, Y.; Yang, Z.; Wang, Z. Risk-sensitive reinforcement learning with function approximation: A debiasing approach. In Proceedings of the International Conference on Machine Learning, Online, 18–24 July 2021; pp. 3198–3207.
19. Jones, M.; Peet, M.M. A generalization of Bellman’s equation with application to path planning, obstacle avoidance and invariant set estimation. *Automatica* **2021**, *127*, 109510. [[CrossRef](#)]
20. Beck, C.; Jentzen, A.; Kleinberg, K.; Kruse, T. Nonlinear Monte Carlo methods with polynomial runtime for Bellman equations of discrete time high-dimensional stochastic optimal control problems. *arXiv* **2023**, arXiv:2303.03390.
21. Beck, C.; Jentzen, A.; Kruse, T. Nonlinear Monte Carlo methods with polynomial runtime for high-dimensional iterated nested expectations. *arXiv* **2020**, arXiv:2009.13989.
22. Becker, S.; Cheridito, P.; Jentzen, A.; Welte, T. Solving high-dimensional optimal stopping problems using deep learning. *Eur. J. Appl. Math.* **2021**, *32*, 470–514. [[CrossRef](#)]
23. Gonon, L. Deep neural network expressivity for optimal stopping problems. *Financ. Stoch.* **2024**, *28*, 865–910. [[CrossRef](#)]
24. Liu, Z.; Sun, C.; Mu, C. An overview on algorithms and applications of deep reinforcement learning. *Chin. J. Intell. Sci. Technol.* **2020**, *2*, 314–326.
25. Kristensen, D.; Mogensen, P.K.; Moon, J.M.; Schjerning, B. Solving dynamic discrete choice models using smoothing and sieve methods. *J. Econom.* **2021**, *223*, 328–360. [[CrossRef](#)]
26. Silver, D.; Veness, J. Monte-Carlo planning in large POMDPs. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 6–9 December 2010; p. 23.
27. Fatica, M.; Phillips, E. Pricing American options with least squares Monte Carlo on GPUs. In Proceedings of the 6th Workshop on High Performance Computational Finance, New York, NY, USA, 18 November 2013; pp. 1–6.
28. Abbas-Turki, L.A.; Vialle, S.; Lapeyre, B.; Mercier, P. Pricing derivatives on graphics processing units using Monte Carlo simulation. *Concurr. Comput. Pract. Exp.* **2014**, *26*, 1679–1697. [[CrossRef](#)]
29. Mao, W.; Zhang, K.; Xie, Q.; Basar, T. Poly-hoot: Monte-carlo planning in continuous space mdps with non-asymptotic analysis. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 4549–4559.
30. Weng, J.; Lin, M.; Huang, S.; Liu, B.; Makoviyichuk, D.; Makoviyichuk, V.; Liu, Z.; Song, Y.; Luo, T.; Jiang, Y.; et al. Envpool: A highly parallel reinforcement learning environment execution engine. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 22409–22421.
31. Cairns, A.J.; Blake, D.; Dowd, K. Stochastic lifestyle: Optimal dynamic asset allocation for defined contribution pension plans. *J. Econ. Dyn. Control* **2006**, *30*, 843–877. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.