

# Accurate and Robust Numerical Methods for the Dynamic Portfolio Management Problem

Fei Cong<sup>1</sup>  · Cornelis W. Oosterlee<sup>1,2</sup>

Accepted: 19 February 2016

© The Author(s) 2016. This article is published with open access at Springerlink.com

**Abstract** This paper enhances a well-known dynamic portfolio management algorithm, the BGSS algorithm, proposed by Brandt et al. (Review of Financial Studies, 18(3):831–873, 2005). We equip this algorithm with the components from a recently developed method, the Stochastic Grid Bundling Method (SGBM), for calculating conditional expectations. When solving the first-order conditions for a portfolio optimum, we implement a Taylor series expansion based on a nonlinear decomposition to approximate the utility functions. In the numerical tests, we show that our algorithm is accurate and robust in approximating the optimal investment strategies, which are generated by a new benchmark approach based on the COS method (Fang and Oosterlee, in SIAM Journal of Scientific Computing, 31(2):826–848, 2008).

**Keywords** Dynamic portfolio management · Simulation method · Least-square regression · Taylor expansion · Fourier cosine expansion method

## 1 Introduction

Solving the dynamic portfolio management problem has become an interesting topic ever since empirical findings in financial research suggested that asset returns were predictable. When the distributions of the asset returns are time-invariant, Merton (1969) and Samuelson (1969) have shown that an investor using a power utility func-

---

✉ Fei Cong  
f.cong@tudelft.nl

Cornelis W. Oosterlee  
c.w.oosterlee@cw.nl

<sup>1</sup> Delft Institute of Applied Mathematics, TU Delft, Delft, The Netherlands

<sup>2</sup> CWI-Centrum Wiskunde & Informatica, Amsterdam, The Netherlands

tion, who re-balances his portfolio optimally, should choose the same asset allocation at every time point, regardless of the investment horizon.

However, if the distributions of the asset returns are time-dependent, for example, when the asset returns follow a vector auto-regression (VAR) model, the optimal asset allocations at intermediate time points are usually not identical. In this case, an investor, who aims to find an optimal asset allocation at initial time, has to first consider all possible asset allocations in subsequent time points, which in turn are also influenced by the initial investment decision. Mathematically, an investor must solve a multivariate optimization problem regarding asset allocations at all portfolio re-balancing times.

Generally it is difficult to solve this multivariate optimization problem directly, and therefore this problem is usually solved by a backward recursion process, where at each time step the investor considers a simplified problem. Basically, there are two stages for solving this step-wise optimization problem. First, we determine how to formulate this optimization problem. We can either focus on the optimization problem directly or try to solve its corresponding first-order conditions, which usually depend on a preparatory approximation of the optimization problem. Then, the rest of the problem can be treated as a mathematical problem of computing conditional expectations. In [Brandt et al. \(2005\)](#), the authors work on the first-order conditions and compute the conditional expectations by simulation and cross-path regression. We call the algorithm the BGSS algorithm. In [van Binsbergen and Brandt \(2007b\)](#), the authors propose an alternative algorithm, the vBB algorithm, where they work on the optimization problem directly via grid-searching but still utilizing simulation and cross-path regression to compute the conditional expectations. They state that the vBB algorithm is more stable than the BGSS algorithm, since the BGSS algorithm essentially relies on an approximation of the utility function. Many other numerical approaches for computing conditional expectations have been considered, for example, in [Barberis \(2000\)](#), [Jondeau and Rockinger \(2006\)](#) and [Garlappi and Skoulakis \(2009\)](#).

In this paper, we propose improvements for the BGSS and the vBB algorithms, which, respectively, rely on solving the first-order conditions and grid-searching to tackle the optimization problem.

In the original BGSS algorithm, cross-path standard regression is employed for solving first-order conditions, which correspond to the utility function via a Taylor series expansion. Within this framework, we particularly contribute in two aspects. First, we replace the standard regression method by the (local) regression combined with bundling of simulation paths, as employed in the stochastic grid bundling method (SGBM), from [Jain and Oosterlee \(2015\)](#). According to our tests, this modification makes the algorithm more stable and robust, and therefore our algorithm performs highly satisfactorily compared to the BGSS and the vBB algorithms, particularly when the investment horizon is long and risk aversion is high. In the process of approximating the utility function, we consider an alternative Taylor expansion to the expansion employed in the original BGSS algorithm. This Taylor expansion was introduced in [Garlappi and Skoulakis \(2009\)](#). This expansion is however not directly compatible with regression-based approaches. With a specific choice of the Taylor expansion center, we can equip our SGBM regression-based portfolio algorithm with this improved Taylor expansion, making the approximations less biased. In short, our enhanced

algorithm still constitutes an algorithm based on simulation and cross-path regression. It thus remains possible to extend this algorithm to high-dimensional scenarios without increasing the computation complexity dramatically.

Based on grid-searching, which is the basic idea of the vBB algorithm, we utilize a Fourier cosine series technique (Fang and Oosterlee 2008, 2009) to compute the conditional expectations and come up with a benchmark algorithm, the COS portfolio management method. Because this method is not based on simulation, there is no such error present in the corresponding numerical results. In the test cases to follow, reference solutions can therefore be generated via this COS-based algorithm.

The paper is organized as follows. Section 2 gives the mathematical formulation of the investor’s problem. In Sect. 2.1, we introduce a special case of the investor’s problem where the simulation- and regression-based methods can be applied. In Sect. 3 the SGBM algorithm is briefly described and the alternative Taylor expansion is also discussed. The benchmark algorithm based on the COS method is presented in Sect. 4. Following that, we display results of the numerical tests in Sect. 5. A brief discussion of the errors of the simulation-based methods is performed in Sect. 5.6. We conclude in the last section.

## 2 Problem Formulation: The Investor’s Problem

This section defines the dynamic portfolio optimization problem, or, in other words, “the investor’s problem”. We assume that the financial market is defined on a complete filtered probability space  $(\Omega, \mathcal{F}, \{\mathcal{F}_t\}_{0 \leq t \leq T}, \mathbb{P})$  with finite time horizon  $[0, T]$ . The state space  $\Omega$  is the set of all realizations of the financial market within the time horizon  $[0, T]$ ,  $\mathcal{F}$  is the sigma algebra of events at time  $T$ , i.e.  $\mathcal{F} = \mathcal{F}_T$ . We assume that the filtration  $\{\mathcal{F}_t\}_{0 \leq t \leq T}$  is generated by the price processes of the financial market and augmented with the null sets of  $\mathcal{F}$ . The probability measure  $\mathbb{P}$  is defined on  $\mathcal{F}$ .

We consider a portfolio consisting of one risk-free asset and  $d$  risky assets, which can be traded at discrete time points,  $t \in [0, 1, \dots, T - 1]$ , before terminal time  $T$ . At each trading time  $t$ , an investor decides his trading strategy to maximize the expected value of the utility of his terminal wealth  $W_T$ . Formally, the investor’s problem is given by

$$V_t(W_t, \mathbf{Z}_t) = \max_{\{\mathbf{x}_s\}_{s=t}^{T-1}} \mathbb{E}[U(W_T) | W_t, \mathbf{Z}_t], \tag{1}$$

subject to the constraints:

$$W_{s+1} = W_s \cdot (\mathbf{x}'_s \mathbf{R}_{s+1}^e + R^f), \quad s = t, \dots, T - 1.$$

Here  $\mathbf{x}_s$  denotes the asset allocation of the investor’s wealth in risky assets. Vector transposition is denoted by the prime sign.  $R^f$  is the return of the risk-free asset, which is assumed to be constant for simplicity, and  $\mathbf{R}_{s+1}^e = [R_{s+1}^{e,1}, \dots, R_{s+1}^{e,d}]$  are the excess returns of the risky assets at time  $s + 1$ . The function  $U(W_T)$  denotes the utility of the investor’s terminal wealth.  $V_t(W_t, \mathbf{Z}_t)$  is termed the *value function*, which measures the investor’s investment opportunities at time  $t$  with wealth  $W_t$  and market state  $\mathbf{Z}_t$ . We assume that  $\{\mathbf{Z}_t\}_{t=0}^T$  is an  $\mathcal{F}_t$ -adapted Markov process.

Mathematically, an investor decides his asset allocations  $\{\mathbf{x}_s\}_{s=0}^{T-1}$  at all time steps to maximize  $V_0(W_0, \mathbf{Z}_0)$  or, equivalently,  $\mathbb{E}[U(W_0 \prod_{s=0}^{T-1} (\mathbf{x}'_s \mathbf{R}_{s+1}^e + R^f)) | W_0, \mathbf{Z}_0]$ .

### 2.1 Numerical Approaches to the Investor’s Problem

From Eq. (1), we see that at time  $t$  it is impossible for the investor to determine the optimal asset allocation  $\mathbf{x}_t$  without knowing optimal asset allocations  $\{\mathbf{x}_s\}_{s=t+1}^{T-1}$  at future time points. A multivariate optimization problem with respect to all asset allocations  $\{\mathbf{x}_s\}_{s=t}^{T-1}$  may be considered, but due to the complexity of the dynamics of  $\mathbf{Z}_t$  it is usually not feasible to solve this problem.

A special case, discussed in Barberis (2000), Brandt et al. (2005) and van Binsbergen and Brandt (2007b), is when the investor has constant relative risk aversion (CRRA)<sup>1</sup>, his optimal asset allocation  $\mathbf{x}_t$  is independent of his wealth  $W_t$ . With this utility function, the optimization problem with respect to the original value function,  $V_t(W_t, \mathbf{Z}_t)$ , which depends on two variables  $W_t$  and  $\mathbf{Z}_t$ , reduces to an optimization problem with respect to a simplified value function,  $v_t(\mathbf{Z}_t)$ :

$$v_t(\mathbf{Z}_t) := V_t(1, \mathbf{Z}_t) = \max_{\{\mathbf{x}_s\}_{s=t}^{T-1}} \mathbb{E} \left[ U \left( \prod_{s=t}^{T-1} (\mathbf{x}'_s \mathbf{R}_{s+1}^e + R^f) \right) \middle| \mathbf{Z}_t \right].$$

Value function  $v_t(\mathbf{Z}_t)$  can be written as a recursive procedure:

$$\begin{aligned} v_t(\mathbf{Z}_t) &= \max_{\{\mathbf{x}_s\}_{s=t}^{T-1}} \mathbb{E} \left[ U \left( \prod_{s=t}^{T-1} (\mathbf{x}'_s \mathbf{R}_{s+1}^e + R^f) \right) \middle| \mathbf{Z}_t \right] \\ &= \max_{\{\mathbf{x}_s\}_{s=t}^{T-1}} \mathbb{E} \left[ \mathbb{E} \left[ U \left( \prod_{s=t}^{T-1} (\mathbf{x}'_s \mathbf{R}_{s+1}^e + R^f) \right) \middle| \mathbf{Z}_{t+1} \right] \middle| \mathbf{Z}_t \right] \\ &= \max_{\mathbf{x}_t} \mathbb{E} \left[ \max_{\{\mathbf{x}_s\}_{s=t+1}^{T-1}} \mathbb{E} \left[ U \left( \prod_{s=t}^{T-1} (\mathbf{x}'_s \mathbf{R}_{s+1}^e + R^f) \right) \middle| \mathbf{Z}_{t+1} \right] \middle| \mathbf{Z}_t \right] \\ &= \max_{\mathbf{x}_t} \mathbb{E} \left[ v_{t+1} \left( (\mathbf{x}'_t \mathbf{R}_{t+1}^e + R^f), \mathbf{Z}_{t+1} \right) \middle| \mathbf{Z}_t \right]. \end{aligned} \tag{4}$$

<sup>1</sup> The CRRA utility function  $U(W_T)$  reads:

$$U(W_T) = \frac{W_T^{1-\gamma}}{1-\gamma}, \quad \gamma \neq 1, \tag{2}$$

and

$$U(W_T) = \log(W_T), \quad \gamma = 1, \tag{3}$$

where Eq. (2) is termed the power utility function and Eq. (3) the log utility function. This utility function is homothetic in wealth, which means that with identical market state  $\mathbf{Z}_t$ , two investors, one with wealth  $W_t$  and the other with wealth 1, will have the same optimal investment strategy at subsequent time points.

Equation (4) is based on the Bellman principle of optimality and dynamic programming (Bellman 1957), which forms the basis for any recursive solution of the dynamic portfolio problem. The principle can be applied since the state vector is assumed to follow a Markov process and, therefore, the optimal asset allocation  $\mathbf{x}_t$  only depends upon time and the current state  $\mathbf{Z}_t$ .

Using the simplified value function and the power utility function with parameter  $\gamma$ , we can solve the investor’s problem, in a backward recursion process<sup>2</sup>, as follows:

- At time  $T$ , we determine the value function as:

$$v_T(\mathbf{Z}_T) = \frac{1}{1 - \gamma}, \quad \gamma \neq 1;$$

- At time  $T - 1$ , the investor considers the optimization problem:

$$\begin{aligned} v_{T-1}(\mathbf{Z}_{T-1}) &= \max_{\mathbf{x}_{T-1}} \mathbb{E} \left[ U \left( \mathbf{x}'_{T-1} \mathbf{R}_T^e + R^f \right) \middle| \mathbf{Z}_{T-1} \right] \\ &= \max_{\mathbf{x}_{T-1}} \mathbb{E} \left[ \left( \mathbf{x}'_{T-1} \mathbf{R}_T^e + R^f \right)^{1-\gamma} v_T(\mathbf{Z}_T) \middle| \mathbf{Z}_{T-1} \right]. \end{aligned}$$

We denote the optimal asset allocation by  $\hat{\mathbf{x}}_{T-1}$ , so that:

$$\max_{\mathbf{x}_{T-1}} \mathbb{E} \left[ U \left( \mathbf{x}'_{T-1} \mathbf{R}_T^e + R^f \right) \middle| \mathbf{Z}_{T-1} \right] := \mathbb{E} \left[ U \left( \hat{\mathbf{x}}'_{T-1} \mathbf{R}_T^e + R^f \right) \middle| \mathbf{Z}_{T-1} \right].$$

Recursively, moving backward in time, the following steps are subsequently performed at times  $t, t = T - 2, T - 3, \dots, 1, 0$ .

- When the investor’s optimal asset allocations,  $\{\hat{\mathbf{x}}_s\}_{s=t+1}^{T-1}$ , are determined, we can calculate the value function  $v_{t+1}(\mathbf{Z}_{t+1})$  as:

$$v_{t+1}(\mathbf{Z}_{t+1}) = \mathbb{E} \left[ U \left( \prod_{s=t+1}^{T-1} \left( \hat{\mathbf{x}}'_s \mathbf{R}_{s+1}^e + R^f \right) \right) \middle| \mathbf{Z}_{t+1} \right].$$

Then, the value function  $v_t(\mathbf{Z}_t)$  reads

$$\begin{aligned} v_t(\mathbf{Z}_t) &= \max_{\{\mathbf{x}_s\}_{s=t}^{T-1}} \mathbb{E} \left[ U \left( \left( \mathbf{x}'_t \mathbf{R}_{t+1}^e + R^f \right) \prod_{s=t+1}^{T-1} \left( \mathbf{x}'_s \mathbf{R}_{s+1}^e + R^f \right) \right) \middle| \mathbf{Z}_t \right] \\ &= \max_{\mathbf{x}_t} \mathbb{E} \left[ \left( \mathbf{x}'_t \mathbf{R}_{t+1}^e + R^f \right)^{1-\gamma} \max_{\{\mathbf{x}_s\}_{s=t+1}^{T-1}} \mathbb{E} \left[ U \left( \prod_{s=t+1}^{T-1} \left( \mathbf{x}'_s \mathbf{R}_{s+1}^e + R^f \right) \right) \middle| \mathbf{Z}_{t+1} \right] \middle| \mathbf{Z}_t \right] \\ &= \max_{\mathbf{x}_t} \mathbb{E} \left[ \left( \mathbf{x}'_t \mathbf{R}_{t+1}^e + R^f \right)^{1-\gamma} v_{t+1}(\mathbf{Z}_{t+1}) \middle| \mathbf{Z}_t \right], \end{aligned} \tag{5}$$

<sup>2</sup> Choosing the CRRA utility function is essential for this process to be valid.

where the last equality is valid by using the definition of  $v_{t+1}(\mathbf{Z}_{t+1})$ . Value function  $v_t(\mathbf{Z}_t)$  can also be written as:

$$\begin{aligned}
 v_t(\mathbf{Z}_t) &= \max_{\mathbf{x}_t} \mathbb{E} \left[ \left( \mathbf{x}'_t \mathbf{R}_{t+1}^e + R^f \right)^{1-\gamma} \mathbb{E} \left[ U \left( \prod_{s=t+1}^{T-1} \left( \hat{\mathbf{x}}'_s \mathbf{R}_{s+1}^e + R^f \right) \right) \middle| \mathbf{Z}_{t+1} \right] \middle| \mathbf{Z}_t \right] \\
 &= \max_{\mathbf{x}_t} \mathbb{E} \left[ \left( \mathbf{x}'_t \mathbf{R}_{t+1}^e + R^f \right)^{1-\gamma} U \left( \prod_{s=t+1}^{T-1} \left( \hat{\mathbf{x}}'_s \mathbf{R}_{s+1}^e + R^f \right) \right) \middle| \mathbf{Z}_t \right], \tag{6}
 \end{aligned}$$

where the last equality follows from the law of iterated expectations.

Either Eqs. (5) or (6) can be employed to evolve the information in the backward recursion. They respectively correspond to the “value function iteration” and the “portfolio weight iteration”, to be discussed in the following subsection. In either case, the optimization problem with respect to  $\mathbf{x}_t$  can be solved via numerical techniques.

As mentioned before, there are basically two numerical approaches available for dealing with this problem, one is by grid-searching and the other is by solving the first-order conditions. These techniques are discussed in subsequent sections.

### Portfolio Weight Iteration or Value Function Iteration

In the backward recursion process, after either the optimal asset allocations  $\{\mathbf{x}_s\}_{s=t+1}^{T-1}$  or  $\mathbf{x}_{t+1}$  and  $v_{t+1}(\mathbf{Z}_{t+1})$  have been determined, we need to evolve the information from time step  $t + 1$  to time step  $t$  to proceed the recursive computation. We can consider either Eqs. (5) or (6) for this purpose. The former is termed “portfolio weight iteration” and the latter “value function iteration”. In [van Binsbergen and Brandt \(2007b\)](#) the authors show that more stable results can be obtained by the portfolio weight iteration. They explain their results as follows. In the value function iteration, the value function is a conditional expectation approximated by cross-path regression and approximation errors may accumulate in the backward recursion process. In the portfolio weight iteration, since the portfolio weights are bounded by borrowing and short-sale constraints, the approximation error remains bounded throughout the whole valuation process.

However, if the value function at each intermediate time step can be approximated accurately, the value function iteration should yield similar results as the portfolio weight iteration. In the numerical tests to follow, we will see that our enhanced numerical methods perform highly satisfactory and in most cases, using the value function iteration produces comparable results as the portfolio weight iteration.

## 3 Solving First-order Conditions

When the value function  $v_{t+1}(\mathbf{Z}_{t+1})$  is known, we consider the optimization problem displayed in Eq. (5).

One approach to obtain the optimal asset allocation  $\mathbf{x}_t$  in Eq. (5) is to solve the first-order conditions for an optimum, i.e.

$$\mathbb{E} \left[ \frac{\partial}{\partial \mathbf{x}_t} \left( (\mathbf{x}'_t \mathbf{R}_{t+1}^e + R^f)^{1-\gamma} v_{t+1}(\mathbf{Z}_{t+1}) \right) \middle| \mathbf{Z}_t \right] = 0. \tag{7}$$

Since Eq. (7) is not directly solvable with respect to  $\mathbf{x}_t$ , in Brandt et al. (2005) the authors proposed an approach to first approximate the value function  $v_t(\mathbf{Z}_t)$  via a Taylor series expansion and then solve the first-order conditions corresponding to the approximated function. Second-order Taylor expansion of the value function is written as<sup>3</sup>:

$$v_t(\mathbf{Z}_t) \approx \max_{\mathbf{x}_t} \left\{ \mathbb{E}[(R^f)^{1-\gamma} v_{t+1}(\mathbf{Z}_{t+1}) | \mathbf{Z}_t] + \mathbb{E}[(1-\gamma)(R^f)^{-\gamma} \mathbf{x}'_t \mathbf{R}_{t+1}^e v_{t+1}(\mathbf{Z}_{t+1}) | \mathbf{Z}_t] + \mathbb{E} \left[ \frac{1}{2} (1-\gamma)(-\gamma)(R^f)^{-1-\gamma} (\mathbf{x}'_t \mathbf{R}_{t+1}^e)^2 v_{t+1}(\mathbf{Z}_{t+1}) | \mathbf{Z}_t \right] \right\}.$$

The corresponding first-order conditions read:

$$\begin{aligned} & \mathbb{E} \left[ (1-\gamma)(R^f)^{-\gamma} \mathbf{R}_{t+1}^e v_{t+1}(\mathbf{Z}_{t+1}) \middle| \mathbf{Z}_t \right] \\ & + \mathbb{E} \left[ (1-\gamma)(-\gamma)(R^f)^{-1-\gamma} (\mathbf{R}_{t+1}^e \mathbf{R}_{t+1}^{e'}) v_{t+1}(\mathbf{Z}_{t+1}) \middle| \mathbf{Z}_t \right] \mathbf{x}_t = 0, \end{aligned}$$

and the optimal asset allocation  $\hat{\mathbf{x}}_t$ , which is assumed to be  $\mathbf{Z}_t$ -measurable, is given by:

$$\hat{\mathbf{x}}_t = \left[ \mathbb{E} \left[ \gamma \cdot (\mathbf{R}_{t+1}^e \mathbf{R}_{t+1}^{e'}) v_{t+1}(\mathbf{Z}_{t+1}) \middle| \mathbf{Z}_t \right] \right]^{-1} \cdot \mathbb{E} \left[ R^f \mathbf{R}_{t+1}^e v_{t+1}(\mathbf{Z}_{t+1}) \middle| \mathbf{Z}_t \right]. \tag{8}$$

Here the conditional expectations can be approximated via simulation and cross-path regression, as done in Brandt et al. (2005), Longstaff and Schwartz (2001) and Tsitsiklis and Roy (2001).

It is mentioned in Brandt et al. (2005) that solving first-order conditions is quite sensitive to the order of the Taylor expansion of the value function and the results from second-order and fourth-order expansions can be different. If we consider the fourth-order Taylor expansion of the value function  $v_t(\mathbf{Z}_t)$ , i.e.

$$v_t(\mathbf{Z}_t) \approx \max_{\mathbf{x}_t} \left\{ \mathbb{E} \left[ (R^f)^{1-\gamma} v_{t+1}(\mathbf{Z}_{t+1}) \middle| \mathbf{Z}_t \right] + \mathbb{E} \left[ (1-\gamma)(R^f)^{-\gamma} \mathbf{x}'_t \mathbf{R}_{t+1}^e v_{t+1}(\mathbf{Z}_{t+1}) \middle| \mathbf{Z}_t \right] + \mathbb{E} \left[ \frac{1}{2} (1-\gamma)(-\gamma)(R^f)^{-1-\gamma} (\mathbf{x}'_t \mathbf{R}_{t+1}^e)^2 v_{t+1}(\mathbf{Z}_{t+1}) \middle| \mathbf{Z}_t \right] \right\}$$

<sup>3</sup> We first consider the Taylor expansion as implemented in Brandt et al. (2005). Another Taylor expansion will be introduced in Sect. 3.2.

$$\begin{aligned}
 &+ \mathbb{E} \left[ \frac{1}{6} (1 - \gamma)(-\gamma)(-1 - \gamma)(R^f)^{-2-\gamma} (\mathbf{x}'_t \mathbf{R}_{t+1}^e)^3 v_{t+1}(\mathbf{Z}_{t+1}) \middle| \mathbf{Z}_t \right] \\
 &+ \mathbb{E} \left[ \frac{1}{24} (1 - \gamma)(-\gamma)(-1 - \gamma)(-2 - \gamma)(R^f)^{-3-\gamma} (\mathbf{x}'_t \mathbf{R}_{t+1}^e)^4 v_{t+1}(\mathbf{Z}_{t+1}) \middle| \mathbf{Z}_t \right] \Bigg\},
 \end{aligned}$$

the optimal asset allocation  $\hat{\mathbf{x}}_t$  is defined as an implicit solution of the following equation:

$$\begin{aligned}
 \hat{\mathbf{x}}_t \approx & \left[ \mathbb{E} \left[ \gamma \cdot (\mathbf{R}_{t+1}^e \mathbf{R}'_{t+1} e) v_{t+1}(\mathbf{Z}_{t+1}) \middle| \mathbf{Z}_t \right] \right]^{-1} \cdot \left\{ \mathbb{E} \left[ R^f \mathbf{R}_{t+1}^e v_{t+1}(\mathbf{Z}_{t+1}) \middle| \mathbf{Z}_t \right] \right. \\
 &+ \frac{1}{2} \mathbb{E} \left[ \frac{(-\gamma)(-1 - \gamma)}{R^f} (\hat{\mathbf{x}}'_t \mathbf{R}_{t+1}^e)^2 \mathbf{R}_{t+1}^e v_{t+1}(\mathbf{Z}_{t+1}) \middle| \mathbf{Z}_t \right] \\
 &\left. + \frac{1}{6} \mathbb{E} \left[ \frac{(-\gamma)(-1 - \gamma)(-2 - \gamma)}{(R^f)^2} (\hat{\mathbf{x}}'_t \mathbf{R}_{t+1}^e)^3 \mathbf{R}_{t+1}^e v_{t+1}(\mathbf{Z}_{t+1}) \middle| \mathbf{Z}_t \right] \right\}. \tag{9}
 \end{aligned}$$

This equation can be treated as a fixed point problem,  $\mathbf{x} = h(\mathbf{x})$  with  $h(\cdot)$  denoting the right-hand side in Eq. (9). This can be solved by an iterative method. To start the iteration, we need an initial guess of the optimal asset allocation. Following the discussion in Brandt et al. (2005), we can take the solution from the second-order Taylor expansion of the value function as the initial guess  $\mathbf{x}_t^0$ .

The iteration can be conducted by Newton’s method for  $h(\mathbf{x}) - \mathbf{x} = 0$ :

$$\mathbf{x}_t^{l+1} = \mathbf{x}_t^l - \frac{h(\mathbf{x}_t^l) - \mathbf{x}_t^l}{h'(\mathbf{x}_t^l) - 1}, \quad l = 0, 1, 2, \dots$$

We stop the iteration, if either the 2-norm of the distance between two consecutive approximations  $\mathbf{x}_t^l$  and  $\mathbf{x}_t^{l+1}$  is smaller than a tolerance value  $\epsilon_{\text{TOL}}$  or the number of iterations reaches a predetermined value  $l_{\text{max}}$ . We take the last iteration  $\mathbf{x}_t^{l+1}$  as the final solution of Eq. (9). In the numerical tests, we choose  $\epsilon_{\text{TOL}} = 0.0001$  and  $l_{\text{max}} = 30$ . Always the tolerance  $\epsilon_{\text{TOL}}$  can be reached, unless stated otherwise.

### 3.1 Stochastic Grid Bundling Method

The Stochastic Grid Bundling Method (SGBM), introduced in Jain and Oosterlee (2015), is a powerful regression-based method for calculating conditional expectations in Eqs. (8) and (9).

It is shown in Jain and Oosterlee (2015) that applying SGBM is highly efficient for obtaining the early-exercise boundary when pricing American-style options and the estimated path-wise option value is so accurate that the Greeks can be generated directly. In Cong and Oosterlee (2016), SGBM is implemented for solving the dynamic mean-variance portfolio management problem in a robust and efficient way. In this paper, we implement SGBM for the dynamic utility-based portfolio management problem. Similar as Brandt et al. (2005), we take the second-order Taylor expansion in the



description of the algorithm for expositional ease. However, in our numerical experiments, we always employ the fourth-order Taylor expansion. Extension to fourth-order expansion can be achieved with the formulas in Eq. (9). Our algorithm can be formally described as follows:

**Step I:** Simulation. Simulate  $N$  paths  $[\mathbf{R}_t^e(i), \mathbf{Z}_t(i)]_{i=1}^N, t = 0, 1, \dots, T$ , and set the value function at terminal time  $T$  as:

$$v_T(\mathbf{Z}_T(i)) = \frac{1}{1 - \gamma}, \quad i = 1, \dots, N.$$

The following steps are subsequently performed at times  $t, t \leq T - 1$ .

**Step II:** Bundling. We bundle the paths at time  $t$  into  $B$  non-overlapping partitions,  $\mathcal{B}_t(1), \dots, \mathcal{B}_t(B)$ . Let each bundle cover a similar number of paths.

**Step III:** Regression. Assume that there are  $N(b)$  paths in bundle  $\mathcal{B}_t(b)$  and their value functions are  $\{v_{t+1}(\mathbf{Z}_{t+1}(i))\}_{i=1}^{N(b)}$  at time  $t + 1$ , or, equivalently, the optimal asset allocations read  $\{\hat{\mathbf{x}}_s(i)\}_{i=1}^{N(b)}, s = t + 1, \dots, N$ , and their excess returns  $\{\mathbf{R}_{t+1}^e(i)\}_{i=1}^{N(b)}$  are known. For these paths, we determine bundle-wise regression parameters  $\{\alpha_k(b)\}_{k=1}^K$  by regressing the values  $\{\gamma \cdot (\mathbf{R}_{t+1}^e(i) \mathbf{R}_{t+1}^{e'}(i)) v_{t+1}(\mathbf{Z}_{t+1}(i))\}_{i=1}^{N(b)}$  on basis functions<sup>4</sup>  $[\phi_1(\mathbf{R}_{t+1}^e(i), \mathbf{Z}_{t+1}(i)), \dots, \phi_K(\mathbf{R}_{t+1}^e(i), \mathbf{Z}_{t+1}(i))\]_{i=1}^{N(b)}$ , which are constructed using the information at time  $t + 1$ . For any path whose state  $\mathbf{Z}_t$  is covered by bundle  $\mathcal{B}_t(b), \mathbb{E}[\gamma \cdot (\mathbf{R}_{t+1}^e \mathbf{R}_{t+1}^{e'}) v_{t+1}(\mathbf{Z}_{t+1}) | \mathbf{Z}_t]$ , the denominator of the right-hand side part in Eq. (8), can be approximated by:

$$\mathbb{E} \left[ \gamma \cdot (\mathbf{R}_{t+1}^e \mathbf{R}_{t+1}^{e'}) v_{t+1}(\mathbf{Z}_{t+1}) \middle| \mathbf{Z}_t \right] \approx \sum_{k=1}^K \alpha_k(b) \mathbb{E} \left[ \phi_k(\mathbf{R}_{t+1}^e, \mathbf{Z}_{t+1}) \middle| \mathbf{Z}_t \right].$$

Similarly,  $\mathbb{E}[R^f \mathbf{R}_{t+1}^e v_{t+1}(\mathbf{Z}_{t+1}) | \mathbf{Z}_t]$ , the numerator of the right-hand side part in Eq. (8), can be approximated by:

$$\mathbb{E} \left[ R^f \mathbf{R}_{t+1}^e v_{t+1}(\mathbf{Z}_{t+1}) \middle| \mathbf{Z}_t \right] \approx \sum_{k=1}^K \beta_k(b) \mathbb{E} \left[ \phi_k(\mathbf{R}_{t+1}^e, \mathbf{Z}_{t+1}) \middle| \mathbf{Z}_t \right],$$

where the regression parameters  $\{\beta_k(b)\}_{k=1}^K$  are obtained by regressing  $\{R^f \mathbf{R}_{t+1}^e(i) v_{t+1}(\mathbf{Z}_{t+1}(i))\}_{i=1}^{N(b)}$  on the same basis functions  $[\phi_1(\mathbf{R}_{t+1}^e(i), \mathbf{Z}_{t+1}(i)), \dots, \phi_K(\mathbf{R}_{t+1}^e(i), \mathbf{Z}_{t+1}(i))\]_{i=1}^{N(b)}$ .

For any path whose state  $\mathbf{Z}_t$  is covered by bundle  $\mathcal{B}_t(b)$ , the optimal asset allocation is approximated by:

<sup>4</sup> In this paper, we always choose for the basis functions basic polynomials.

$$\hat{\mathbf{x}}_t \approx \left[ \sum_{k=1}^K \alpha_k(b) \mathbb{E} \left[ \phi_k(\mathbf{R}_{t+1}^e, \mathbf{Z}_{t+1}) \middle| \mathbf{Z}_t \right] \right]^{-1} \cdot \left[ \sum_{k=1}^K \beta_k(b) \mathbb{E}[\phi_k(\mathbf{R}_{t+1}^e, \mathbf{Z}_{t+1}) \middle| \mathbf{Z}_t] \right].$$

The regression step is repeated for all bundles at each time step, so for each path we find the corresponding optimal asset allocation.

**Step IV:** Transition. For the  $i$ -th path in bundle  $\mathcal{B}_t(b)$ , we can either apply portfolio weight iteration or value function iteration to transfer the information of the optimal investment strategy from time  $t$  to time  $t - 1$ .

When using the portfolio weight iteration, we just store the optimal asset allocations  $\{\mathbf{x}_s\}_{s=t}^{T-1}$  and write  $v_t(\mathbf{Z}_t)$  as  $\prod_{s=t}^{T-1} (\mathbf{x}'_s \mathbf{R}_{s+1}^e + R^f)^{1-\gamma} / (1 - \gamma)$  in the regression step at time  $t - 1$ .

If we use the value function iteration, the process is slightly more involved. For all paths in bundle  $\mathcal{B}_t(b)$ , we regress  $\{(x'_t(i) \mathbf{R}_{t+1}^e(i) + R^f)^{1-\gamma} v_{t+1}(\mathbf{Z}_{t+1}(i))\}_{i=1}^{N(b)}$  on the following polynomial basis functions  $[\hat{\phi}_1(\mathbf{Z}_t(i)), \dots, \hat{\phi}_K(\mathbf{Z}_t(i))\]_{i=1}^{N(b)}$  formed by  $\{\mathbf{Z}_t(i)\}_{i=1}^{N(b)}$  and obtain regression parameters  $[\varphi_1(b), \dots, \varphi_K(b)]$ . The value function is then approximated by<sup>5</sup>:

$$v_t(\mathbf{Z}_t) \approx \sum_{k=1}^K \varphi_k(b) \hat{\phi}_k(\mathbf{Z}_t).$$

### 3.2 Taylor Expansion Based on a Nonlinear Decomposition

In both, the BGSS and SGBM:<sup>6</sup> algorithms, an essential step before solving the equations for the first-order conditions is to rewrite the value function,  $v_t(\mathbf{Z}_t)$ , in a Taylor series expansion in which the asset allocation  $\mathbf{x}_t$  is separated from the conditional expectations of  $\mathbf{R}_{t+1}^e$ . A  $K$ th-order Taylor expansion in SGBM can be written as<sup>7</sup>

$$\left(x_t R_{t+1}^e + R^f\right)^{1-\gamma} \approx \sum_{k=0}^K \frac{g_1^{(k)}(0)}{k!} (x_t R_{t+1}^e)^k, \tag{10}$$

where  $g_1^{(k)}(0)$  denotes the  $k$ th derivative of function  $g_1(y)$  when  $y = 0$ . Function  $g_1(y) = (y + R^f)^{1-\gamma}$ . So,  $g_1(x_t R_{t+1}^e) = (x_t R_{t+1}^e + R^f)^{1-\gamma}$ .

Since the excess return  $R_t^e$  is a nonlinear transformation of the log excess return  $\mathbf{r}_t^e$ , i.e.

$$R_{t+1}^e = \exp(r_{t+1}^e) R^f - R^f,$$

<sup>5</sup> It should be noted that the standard regression method is implemented in this step. According to our tests, introducing SGBM in this step is not helpful.

<sup>6</sup> With a little abuse of the term, we also denote our dynamic portfolio management method by ‘‘SGBM’’.

<sup>7</sup> For simplicity, we describe the dynamic portfolio management problem with one risky asset here.

an alternative way to perform a Taylor expansion for  $(x_t R_{t+1}^e + R^f)^{1-\gamma}$  is given by:

$$\left(x_t R_{t+1}^e + R^f\right)^{1-\gamma} \approx \sum_{k=0}^K \frac{g_2^{(k)}(0)}{k!} (r_{t+1}^e)^k, \tag{11}$$

where function  $g_2(z)$  is defined by:

$$g_2(z) = \left(R^f + x_t \left(\exp(z) R^f - R^f\right)\right)^{1-\gamma}.$$

Functions  $g_1(x_t R_{t+1}^e)$  and  $g_2(r_t^e)$  are both identical to  $(x_t R_{t+1}^e + R^f)^{1-\gamma}$ , but different ways of choosing the underlying variable yield different Taylor expansion formulas.

In [Garlappi and Skoulakis \(2011\)](#), the authors term the expansion described in Eq. (10) as ‘‘Taylor expansion based on a linear decomposition’’ and the expansion described in Eq. (11) as ‘‘Taylor expansion based on a nonlinear decomposition’’. They show that when the centers of Taylor expansions are carefully chosen, the ‘‘Taylor expansion based on a nonlinear decomposition’’ is more accurate than the ‘‘Taylor expansion based on a linear decomposition’’ when approximating the function  $(x_t R_{t+1}^e + R^f)^{1-\gamma}$ . We will call these expansions ‘‘original Taylor’’ and ‘‘log Taylor’’ expansions, respectively, in the rest of this paper. Although the log Taylor expansion has been implemented in [Garlappi and Skoulakis \(2009\)](#) for dynamic portfolio management, their choice of expansion center is not compatible with the algorithm discussed here. We deal with this problem by performing a log Taylor expansion around center 0, as displayed in Eq. (11).<sup>8</sup> According to the numerical tests in Sect. 5, we find that the log Taylor expansion is indeed a superior choice even when the expansion center is chosen to be 0. The reasoning is that the log excess return  $r_t^e$  usually exhibits a distribution similar to the normal distribution. Therefore, a Taylor expansion with respect to this variable, i.e. the so-called ‘‘log Taylor’’ expansion, can yield accurate results with a limited number of expansion terms. The distribution of the excess return  $R_t^e$  usually exhibits a fat tailed distribution, which requires more terms in the original Taylor expansion to approximate its moments.

### 4 Grid-Searching Methods

An alternative technique to solving first-order conditions is based on grid-searching, which is an intuitive idea for solving the optimization problem described in Eq. (5).

<sup>8</sup> We especially choose the Taylor expansion center to be 0, because only in this case is there no term related to  $x_t$  inside the power transformation, for example:

$$\begin{aligned} g_2(0) &= (R^f)^{1-\gamma}, \\ g_2'(0) &= (1 - \gamma)(R^f)^{1-\gamma} x_t, \\ g_2''(0) &= (1 - \gamma)(R^f)^{1-\gamma} (-\gamma x_t^2 + x_t). \end{aligned}$$

In grid-searching, we reduce the optimization problem on the continuous domain to a problem on a discrete domain. For example, if we consider the allocation,  $x_t$ , of one risky asset, the original optimization problem is solved on a domain  $[0, 1]$ . By grid-searching, we construct  $M$  equidistant grid points  $\{\frac{m}{M}\}_{m=0}^M$  and consider the optimization problem on the discrete domain  $D_M = \{\frac{m}{M} \mid m = 0, 1, \dots, M\}$ . To solve this discrete optimization problem, we test each possible choice of the allocation  $x_t^{(m)} = \frac{m}{M}, m = 0, \dots, M$  and calculate the corresponding value functions:

$$v_t^{(m)}(\mathbf{Z}_t) = \mathbb{E} \left[ (x_t^{(m)} R_{t+1}^e + R^f)^{1-\gamma} v_{t+1}(\mathbf{Z}_{t+1}) \mid \mathbf{Z}_t \right]. \tag{12}$$

We determine the maximum,  $v_t^{\max}(\mathbf{Z}_t)$ , from  $\{v_t^{(m)}(\mathbf{Z}_t)\}_{m=0}^M$  and denote its corresponding asset allocation as “the optimal asset allocation”.

Although it is mentioned in [van Binsbergen and Brandt \(2007a, b\)](#) that the grid-searching method is robust and avoids a number of numerical issues regarding convergence that occur when solving first-order conditions, it should be noted that grid-searching is an expensive numerical approach. The workload of grid-searching grows exponentially as the dimensionality of the problem increases. Moreover, according to our numerical tests in the low-dimensional cases, the vBB algorithm, which employs grid-searching together with simulation, yields more “uncertain” results (larger variance) compared to the other simulation-based algorithms.

However, if we wish to find an accurate reference solution to the dynamic portfolio management problem, grid-searching seems our only choice since solving first-order conditions essentially relies on Taylor approximations of the utility function, whereas grid-searching does not. In the next subsection we will present our benchmark approach based on the idea of grid-searching.

### 4.1 COS Portfolio Management Method

In this section, we present a benchmark method, based on the Fourier cosine series expansion (COS) method to calculate the conditional expectations. This method was introduced in [Fang and Oosterlee \(2008\)](#) for pricing one-dimensional European options and later in [Fang and Oosterlee \(2009\)](#) for pricing one-dimensional Bermudan and barrier options. In [Ruijter and Oosterlee \(2012\)](#), this method was extended to the two-dimensional case. Because the COS method is not based on simulation, it can yield benchmark solutions to the investor’s problem, especially in the basic case with one risky asset and one risk-free asset. Following the previous discussions, this basic investor’s problem with power utility function is given by:

$$v_{t-1}(\mathbf{Z}_{t-1}) = \max_{x_{t-1}} \mathbb{E} \left[ (x_{t-1} R_t^e + R^f)^{1-\gamma} v_t(\mathbf{Z}_t) \mid \mathbf{Z}_{t-1} \right] \quad t = 1, \dots, T - 1, \tag{13}$$

where the terminal condition reads:

$$v_T(\mathbf{Z}_T) = \frac{1}{1-\gamma}, \quad \gamma \neq 1.$$

If we denote the conditional transition density function from state  $Z_{t-1}$  to  $(R_t^e, Z_t)$  as  $f(R_t^e, Z_t|Z_{t-1})$ , the investor’s problem reads:

$$v_{t-1}(Z_{t-1}) = \max_{x_{t-1}} \iint_{\mathbb{R}^2} (x_{t-1}R_t^e + R^f)^{1-\gamma} v_t(Z_t) f(R_t^e, Z_t|Z_{t-1}) dR_t^e dZ_t, \quad t = 1, \dots, T - 1. \tag{14}$$

The COS algorithm for calculating conditional expectations can be described in five steps:

**Step I:** Truncate the integration range in Eq. (14).

If we assume that the integrand is integrable, we can truncate the integration range from  $\mathbb{R}^2$  to  $[a_R, b_R] \times [a_Z, b_Z]$  without losing significant accuracy. The approximated value function  $\hat{v}_{t-1}(Z_{t-1})$  reads:

$$\hat{v}_{t-1}(Z_{t-1}) = \max_{x_{t-1}} \int_{a_Z}^{b_Z} \int_{a_R}^{b_R} (x_{t-1}R_t^e + R^f)^{1-\gamma} v_t(Z_t) f(R_t^e, Z_t|Z_{t-1}) dR_t^e dZ_t.$$

*Remark 4.1* For one variable, for example  $Z_t$ , the suggested integration range  $[a_Z, b_Z]$  in Fang and Oosterlee (2009) and Ruijter and Oosterlee (2012) is  $[\xi_1^Z - L\xi_2^Z, \xi_1^Z + L\xi_2^Z]$ , where  $\xi_1^Z$  is the mean of  $Z_t$  and  $\xi_2^Z$  the standard deviation of  $Z_t$ .  $L$  should be large enough to make the truncation error acceptably low.

**Step II:** Expand the integrand in Fourier cosines.

If we denote the Fourier cosine expansion of  $f(R_t^e, Z_t|Z_{t-1})$  on  $[a_R, b_R] \times [a_Z, b_Z]$  by:

$$A_{k_1, k_2}(Z_{t-1}) := \frac{2}{b_Z - a_Z} \frac{2}{b_R - a_R} \int_{a_Z}^{b_Z} \int_{a_R}^{b_R} f(R_t^e, Z_t|Z_{t-1}) \times \cos\left(k_1\pi \frac{R_t^e - a_R}{b_R - a_R}\right) \cos\left(k_2\pi \frac{Z_t - a_Z}{b_Z - a_Z}\right) dR_t^e dZ_t,$$

and similarly define the utility coefficients as:

$$V_{k_1, k_2}(t, x_{t-1}) := \frac{2}{b_Z - a_Z} \frac{2}{b_R - a_R} \int_{a_Z}^{b_Z} \int_{a_R}^{b_R} (x_{t-1}R_t^e + R^f)^{1-\gamma} v_t(Z_t) \times \cos\left(k_1\pi \frac{R_t^e - a_R}{b_R - a_R}\right) \cos\left(k_2\pi \frac{Z_t - a_Z}{b_Z - a_Z}\right) dR_t^e dZ_t,$$

value function  $v_{t-1}(Z_{t-1})$  can be approximated by:

$$\hat{v}_{t-1}(Z_{t-1}) = \max_{x_{t-1}} \left\{ \frac{b_Z - a_Z}{2} \frac{b_R - a_R}{2} \sum'_{k_1=0} \sum'_{k_2=0} A_{k_1, k_2}(Z_{t-1}) V_{k_1, k_2}(t, x_{t-1}) \right\}.$$

The primed sum  $\sum'$  means that the first term of the summation has half weight.

**Step III:** Truncate the infinite series.

We truncate the infinite series, as follows:

$$\bar{v}_{t-1}(Z_{t-1}) = \max_{x_{t-1}} \left\{ \frac{b_Z - a_Z}{2} \frac{b_R - a_R}{2} \sum_{k_1=0}^{N_1-1} \sum_{k_2=0}^{N_2-1} A_{k_1,k_2}(Z_{t-1}) V_{k_1,k_2}(t, x_{t-1}) \right\}.$$

**Step IV:** Calculate the coefficients  $A_{k_1,k_2}(Z_{t-1})$ .

The coefficients  $A_{k_1,k_2}(Z_{t-1})$  can be approximated by  $F_{k_1,k_2}(Z_{t-1})$ , as follows:

$$F_{k_1,k_2}(Z_{t-1}) := \frac{2}{b_Z - a_Z} \frac{2}{b_R - a_R} \iint_{\mathbb{R}^2} f(R_t^e, Z_t | Z_{t-1}) \times \cos\left(k_1\pi \frac{R_t^e - a_R}{b_R - a_R}\right) \cos\left(k_2\pi \frac{Z_t - a_Z}{b_Z - a_Z}\right) dR_t^e dZ_t.$$

Using the following property of cosines:  $2 \cos(\alpha) \cos(\beta) = \cos(\alpha + \beta) + \cos(\alpha - \beta)$ , we can calculate  $F_{k_1,k_2}(Z_{t-1})$  by:

$$F_{k_1,k_2}(Z_{t-1}) = \frac{F_{k_1,k_2}^+(Z_{t-1}) + F_{k_1,k_2}^-(Z_{t-1})}{2},$$

where

$$\begin{aligned} & F_{k_1,k_2}^\pm(Z_{t-1}) \\ &= \frac{2}{b_Z - a_Z} \frac{2}{b_R - a_R} \iint_{\mathbb{R}^2} f(R_t^e, Z_t | Z_{t-1}) \cos\left(k_1\pi \frac{R_t^e - a_R}{b_R - a_R} \pm k_2\pi \frac{Z_t - a_Z}{b_Z - a_Z}\right) dR_t^e dZ_t \\ &= \frac{2}{b_Z - a_Z} \frac{2}{b_R - a_R} \Re\left(\iint_{\mathbb{R}^2} f(R_t^e, Z_t | Z_{t-1}) \exp\left(ik_1\pi \frac{R_t^e}{b_R - a_R} \pm ik_2\pi \frac{Z_t}{b_Z - a_Z}\right) dR_t^e dZ_t\right. \\ &\quad \left. \times \exp\left(-ik_1\pi \frac{a_R}{b_R - a_R} \mp ik_2\pi \frac{a_Z}{b_Z - a_Z}\right)\right) \\ &= \frac{2}{b_Z - a_Z} \frac{2}{b_R - a_R} \Re\left(\psi\left(\frac{k_1\pi}{b_R - a_R}, \pm \frac{k_2\pi}{b_Z - a_Z} \middle| Z_{t-1}\right)\right. \\ &\quad \left. \times \exp\left(-ik_1\pi \frac{a_R}{b_R - a_R} \mp ik_2\pi \frac{a_Z}{b_Z - a_Z}\right)\right). \end{aligned}$$

$\Re(\cdot)$  means taking the real part of the input data.  $\psi(u_r, u_Z | Z_{t-1})$  is the *bivariate conditional characteristic function* of  $(R_t^e, Z_t)$  given state  $Z_{t-1}$ :

$$\psi(u_R, u_Z | Z_{t-1}) = \iint_{\mathbb{R}^2} \exp(i[u_R, u_Z] \cdot [R_t^e, Z_t]') f(R_t^e, Z_t | Z_{t-1}) dR_t^e dZ_t.$$

For many asset dynamics models this bivariate characteristic function is known in closed form.

**Step V:** Calculate the coefficients  $V_{k_1,k_2}(t, x_{t-1})$ .

The coefficients  $V_{k_1,k_2}(t, x_{t-1})$  are not directly related to any closed-form expression. However, we can apply numerical integration and the discrete cosine transform (DCT)

to approximate  $V_{k_1, k_2}(t, x_{t-1})$ . To do this, we take  $Q \geq \max[N_1, N_2]$  grid points in each spatial dimension and define:

$$\begin{aligned} R_t^{n_1} &:= a_R + (n_1 + \frac{1}{2})\Delta R_t \quad n_1 = 1, \dots, Q \\ Z_t^{n_2} &:= a_Z + (n_2 + \frac{1}{2})\Delta Z_t \quad n_2 = 1, \dots, Q \\ \Delta R_t &:= \frac{b_R - a_R}{Q}, \quad \Delta Z_t := \frac{b_Z - a_Z}{Q}. \end{aligned}$$

The midpoint-rule integration gives us

$$\begin{aligned} V_{k_1, k_2}(t, x_{t-1}) &\approx \sum_{n_1=0}^{Q-1} \sum_{n_2=0}^{Q-1} \frac{2}{b_R - a_R} \frac{2}{b_Z - a_Z} \left(x_{t-1} R_t^e + R^f\right)^{1-\gamma} v_t(Z_t) \\ &\times \cos\left(k_1\pi \frac{R_t^{n_1} - a_R}{b_R - a_R}\right) \cos\left(k_2\pi \frac{Z_t^{n_2} - a_Z}{b_Z - a_Z}\right) \Delta R_t \Delta Z_t \\ &= \sum_{n_1=0}^{Q-1} \sum_{n_2=0}^{Q-1} \frac{2}{Q} \frac{2}{Q} \left(x_{t-1} R_t^{n_1} + R^f\right)^{1-\gamma} v_t(Z_t^{n_2}) \cos\left(k_1\pi \frac{R_t^{n_1} - a_R}{b_R - a_R}\right) \\ &\times \cos\left(k_2\pi \frac{Z_t^{n_2} - a_Z}{b_Z - a_Z}\right). \end{aligned}$$

The equation above can be calculated efficiently via a two-dimensional DCT, for example, with the function `dct2` of MATLAB. Moreover, we can rewrite the sum of multiplications into a multiplication of sums, that is:

$$\begin{aligned} V_{k_1, k_2}(t, x_{t-1}) &\approx \frac{2}{Q} \frac{2}{Q} \left( \sum_{n_1=0}^{Q-1} \left(x_{t-1} R_t^{n_1} + R^f\right)^{1-\gamma} \cos\left(k_1\pi \frac{R_t^{n_1} - a_R}{b_R - a_R}\right) \right) \\ &\times \left( \sum_{n_2=0}^{Q-1} v_t(Z_t^{n_2}) \cos\left(k_2\pi \frac{Z_t^{n_2} - a_Z}{b_Z - a_Z}\right) \right). \end{aligned}$$

Then, the two-dimensional DCT can be replaced by two separate one-dimensional DCTs, which helps reducing the computational time.

For state  $Z_{t-1}$  and asset allocation  $x_{t-1}$ , we can calculate the conditional expectation shown in Eq. (13) by the COS method. To solve the optimization problem with respect to  $x_{t-1}$ , we employ grid-searching: we evaluate discretized values of  $x_{t-1} \in \{\frac{m}{M} | m = 0, \dots, M\}$  and find the largest conditional expectation. The backward recursion process can be performed from time  $T - 1$  to the initial time.

Within the COS method, we have five parameters to adjust the truncation and discretization errors. These are  $N_1, N_2, L, Q$  and  $M$ . Generally, larger values of these parameters lead to more accurate approximations but also to higher computational load. We use the following default parameter setting:

$$N_1 = 50, \quad N_2 = 100, \quad L = 8, \quad Q = 100, \quad M = 200. \quad (15)$$

According to our experiments, the COS method provides highly accurate results under this setting. However, when the admissible asset allocation can be chosen from a very wide range of values, the COS approach, which is based on discrete grid search, may lose its accuracy. In that case, the SGBM method equipped with the log Taylor expansion and a large number of paths will still generate satisfactory solutions and appears favorable.

*Remark 4.2* The COS method suffers from the curse of dimensionality. However, this is a problem for any method involving discretization of the state space and grid-searching. To settle this issue in high-dimensional cases, adaptive discretization, or sparse grids, and grid-searching can be applied.

*Remark 4.3* The computational load of the COS method for a dynamic portfolio management problem is mainly related to the DCT computations, for which the computational complexity at each time step is  $O(N_2 \cdot M \cdot Q \cdot \log(Q))$ . Computations at each time step are performed sequentially, but the computations for the value function at each state point are independent, so it should be possible to accelerate the COS method by parallel processing.

## 5 Numerical Experiments

In this section, we test the performance of five methods for generating the optimal dynamic portfolio management strategy. These are:

- “BGSS”: the method introduced in [Brandt et al. \(2005\)](#);
- “vBB”: the method introduced in [van Binsbergen and Brandt \(2007b\)](#);
- “SGBM”: SGBM with the original Taylor expansion;
- “SGBM-LT”: SGBM with the log Taylor expansion;
- “COS”: the COS method.

We impose borrowing and short-sale constraints on the asset allocations, that are therefore restricted between 0 and 1. When we implement the simulation-based algorithms, *we always generate  $2^{14}$  paths*. For “SGBM” and “SGBM-LT”, which require bundling, *we employ 32 bundles at each time step*. We approximate the utility function by Taylor expansions, up to 4th-order for both the log Taylor expansion and the original Taylor expansion. For “BGSS” and “vBB”, we use polynomials of the state variable *up to second-order as the basis functions for the cross-path regression*. For “SGBM” and “SGBM-LT”, the polynomials are also second-order but in of the state variable *and* the return variable.

To measure the performance of a dynamic portfolio management strategy, we consider the statistic, “annualized certainty equivalent rate”, CER. It describes the annualized return rate of a risk-free asset which at terminal time  $Y$ (years) yields the same utility of wealth obtained from the dynamic portfolio management strategy. Equivalently, the CER is the risk-free rate that an investor is willing to accept rather



than adopting a particular risky portfolio management strategy. Formally the CER is defined by:

$$U(W_0 \cdot (1 + \text{CER})^Y) = V_0(W_0, \mathbf{Z}_0), \tag{16}$$

where the value function  $V_0(W_0, \mathbf{Z}_0)$  is defined by Eq. (1). Generally, a portfolio management strategy with high CER is close to the optimal strategy and can thus be regarded as an accurate solution to the dynamic portfolio management problem.

We perform numerical tests here for a basic test case where the portfolio contains one risky asset and one risk-free asset. We consider the vector auto-regression (VAR) model to describe the dynamics of the log excess return  $r_t^e$  of the risky asset and its log dividend yield  $d_t$ , that are chosen as the state variables. Quarterly data are generated with the following process, as in [Brandt et al. \(2005\)](#), [van Binsbergen and Brandt \(2007b\)](#) and [Garlappi and Skoulakis \(2009\)](#):

$$\begin{bmatrix} r_{t+1}^e \\ d_{t+1} \end{bmatrix} = \begin{bmatrix} 0.227 \\ -0.155 \end{bmatrix} + \begin{bmatrix} 0.060 \\ 0.958 \end{bmatrix} d_t + \begin{bmatrix} \epsilon_{t+1}^r \\ \epsilon_{t+1}^d \end{bmatrix},$$

where

$$\begin{bmatrix} \epsilon_{t+1}^r \\ \epsilon_{t+1}^d \end{bmatrix} \sim N(\mu_\epsilon, \Sigma_\epsilon), \quad \mu_\epsilon = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad \text{and} \quad \Sigma_\epsilon = \begin{bmatrix} 0.0060 & -0.0051 \\ -0.0051 & 0.0049 \end{bmatrix}.$$

In most of the tests, the initial state,  $d_0$ , is chosen as the unconditional mean, i.e.,  $d_0 = -0.155 / (1 - 0.958) = -3.6905$ . Only in Sect. 5.4 we will consider three quantiles, the 25, 50 and 75 % quantiles, of the unconditional distribution of state variable respectively as the initial state. The gross return of the risk-free asset is chosen as  $R^f = 1.06^{0.25}$  and the excess return  $R_t^e$  of the risky asset is  $R_t^e = R^f (\exp(r_t^e) - 1)$ .

Associated to the 1D-VAR model, the characteristic function, which is essential for the COS portfolio management method, can be formulated as<sup>9</sup>:

$$\begin{aligned} \psi(u_r, u_Z | Z_{t-1}) &= \exp \left( ik_1 \pi \frac{0.227 + 0.060 \cdot Z_{t-1}}{b_r - a_r} + ik_2 \pi \frac{-0.155 + 0.958 \cdot Z_{t-1}}{b_Z - a_Z} \right) \\ &\times \exp \left( i \mu_\epsilon' [u_r, u_Z]' - \frac{1}{2} [u_r, u_Z] \Sigma_\epsilon [u_r, u_Z]' \right). \end{aligned}$$

### 5.1 Quality of the COS Portfolio Management Method

We first check the validity and quality of the COS portfolio management method. For the dynamic portfolio management problem with the 1D-VAR model, we calculate the optimal asset allocations and the corresponding annualized certainty equivalent rates and compare them with the reference values from [Garlappi and Skoulakis \(2009\)](#). As we can see in Table 1, in case of different investment horizons and risk aversions,

<sup>9</sup> Since  $R_t^e$  is an injective function of  $r_t^e$ , for all equations in Sect. 4 replacing the functions of  $R_t^e$  by functions of  $r_t^e$  is valid with trivial modification.

**Table 1** Initial optimal asset allocations and the corresponding annualized certainty equivalent rates of the COS portfolio management method, based on the 1D-VAR model, with reference values from [Garlappi and Skoulakis \(2009\)](#)

|          | Optimal initial asset allocation (%) |           |               |           | Annualized CER (%) |           |               |           |
|----------|--------------------------------------|-----------|---------------|-----------|--------------------|-----------|---------------|-----------|
|          | $\gamma = 5$                         |           | $\gamma = 15$ |           | $\gamma = 5$       |           | $\gamma = 15$ |           |
|          | COS                                  | Reference | COS           | Reference | COS                | Reference | COS           | Reference |
| $T = 10$ | 41.5                                 | 42.8      | 15.0          | 15.6      | 7.23               | 7.22      | 6.43          | 6.43      |
| $T = 20$ | 56.5                                 | 56.3      | 23.0          | 25.3      | 7.84               | 7.84      | 6.72          | 6.72      |
| $T = 30$ | 69.0                                 | 66.9      | 33.0          | 35.4      | 8.26               | 8.26      | 7.01          | 7.01      |
| $T = 40$ | 77.5                                 | 76.8      | 43.0          | 44.5      | 8.53               | 8.53      | 7.27          | 7.26      |

the COS method always provides accurate approximations of the annualized certainty equivalent rates and also highly satisfactory approximations of the optimal initial asset allocations.

As the COS method with the parameter settings in (15) and the reference method involve some approximation errors, it is difficult to say whose optimal initial asset allocation is superior. However, since it is known that first-order deviations in the portfolio policy have only second-order welfare effects (Cochrane 1989) and the COS method and the reference method yield similar annualized certainty equivalent rates, we consider these as the optimal solutions when comparing with simulation-based methods.

*Remark 5.1* We have also tested the performance of the COS portfolio management method with different initial states  $d_0$ . For any initial state tested, it generates very similar results as the reference values in [Garlappi and Skoulakis \(2009\)](#).

### 5.2 Portfolio Management with the Buy-and-Hold Strategy

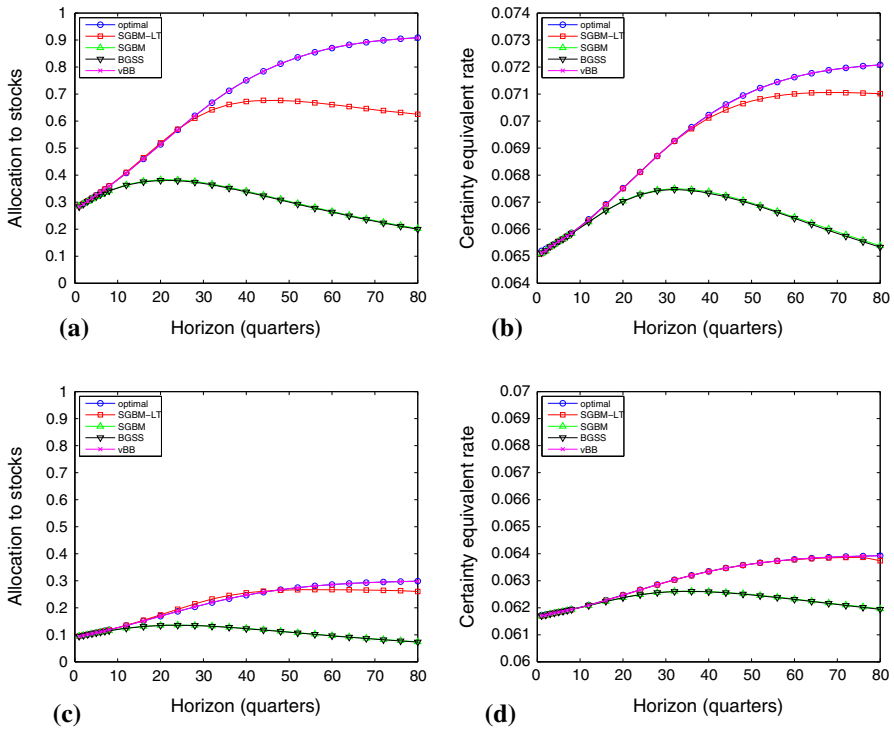
In this section, instead of the dynamic portfolio management problem, in which an investor decides his optimal asset allocations at intermediate times  $t = 0, 1, \dots, T - 1$ , we consider a case where the investor decides his optimal asset allocation at time  $t = 0$  and holds a fixed amount of assets until terminal time  $t = T$ . The corresponding value function reads

$$v_0(Z_0) = \max_{x_0} \mathbb{E} \left[ \frac{1}{1 - \gamma} \left( x_0 R_{0 \rightarrow T}^e + R_{0 \rightarrow T}^f \right)^{1-\gamma} \middle| Z_0 \right],$$

where  $R_{0 \rightarrow T}^f = (R^f)^T$ ,  $R_{0 \rightarrow T}^e = R_{0 \rightarrow T}^f \cdot e^{\sum_{i=1}^T r_i^e} - R_{0 \rightarrow T}^f$ .

This type of problem can be viewed as a static portfolio management problem, for which the aforementioned four simulation-based methods (“SGBM-LT”, “SGBM”, “BGSS” and “vBB”) can be applied. The COS method is utilized to generate benchmarks for the optimal asset allocations and the corresponding annualized certainty equivalent rates.

Figure 1 shows that “vBB” provides identical results to the optimal ones, since it does not involve Taylor expansion errors. For the other three candidates, in which



**Fig. 1** For the simulation-based methods, we report the point estimate of the initial asset allocations from 100 runs. The optimal values are generated with the COS method, **a** optimal allocation  $\gamma = 5$ , **b** certainty equivalent rate  $\gamma = 5$ , **c** optimal allocation  $\gamma = 15$ , **d** certainty equivalent rate  $\gamma = 15$

Taylor expansions are involved, “SGBM-LT” provides the best approximation of the initial asset allocations. When the investment horizon is long, although the asset allocations of “SGBM-LT” are not close to the optimal solutions, their corresponding certainty equivalent rates are similar to the optimal ones. For the other two methods, “SGBM” and “BGSS”, the estimates of asset allocations and certainty equivalent rates are acceptable only when the investment horizon is shorter than 10 quarters.

This test indicates that the log Taylor expansion ( $2^{14}$  paths, 32 bundles) outperforms the original Taylor expansion for approximating the utility functions. The advantage of using the log Taylor expansion is obvious when the distribution of the accumulated excess return,  $R_{0 \rightarrow T}^e$ , exhibits a fat tail.

### 5.3 Dynamic Portfolio Management with Different Investment Horizons and Risk Aversion Parameters

Following the discussion in [van Binsbergen and Brandt \(2007b\)](#), we consider for the dynamic optimization problem the portfolio weight iteration in the transfer step and compare the four simulation-based methods.

**Table 2** Mean and the standard derivations of the CER from 100 runs, comparing 4 simulation-based methods for dynamic portfolio management for different investment horizons and risk aversion parameters

| Annualized certainty equivalent rate (%) |      |        |      |        |      |        |         |        |      |
|--|------|--------|------|--------|------|--------|---------|--------|------|
| $\gamma$                                 | BGSS | (SE)   | vBB  | (SE)   | SGBM | (SE)   | SGBM-LT | (SE)   | COS  |
| <i>T</i> = 40                            |      |        |      |        |      |        |         |        |      |
| 5  | 8.51 | (0.02) | 8.52 | (0.02) | 8.55 | (0.02) | 8.56    | (0.02) | 8.53 |
| 10                                       | 7.61 | (0.03) | 7.63 | (0.04) | 7.67 | (0.02) | 7.77    | (0.03) | 7.74 |
| 15                                       | 7.12 | (0.03) | 7.08 | (0.05) | 7.17 | (0.02) | 7.26    | (0.02) | 7.27 |
| 20                                       | 6.85 | (0.03) | 6.72 | (0.06) | 6.90 | (0.01) | 6.97    | (0.02) | 6.98 |
| <i>T</i> = 80                            |      |        |      |        |      |        |         |        |      |
| 5  | 8.79 | (0.02) | 8.78 | (0.02) | 8.96 | (0.01) | 8.97    | (0.01) | 8.94 |
| 10                                       | 7.76 | (0.04) | 7.63 | (0.03) | 8.08 | (0.02) | 8.28    | (0.02) | 8.29 |
| 15                                       | 7.18 | (0.05) | 7.03 | (0.04) | 7.47 | (0.01) | 7.65    | (0.02) | 7.83 |
| 20                                       | 6.86 | (0.05) | 6.65 | (0.06) | 7.13 | (0.01) | 7.28    | (0.01) | 7.49 |

The portfolio weight iteration is utilized. The COS method serves as the reference

In Table 2, we observe that “SGBM-LT”, among the four methods, always provides the highest certainty equivalent rates, which implies that the portfolio management strategy generated by “SGBM-LT” is most similar to the optimal one. However, when the investment horizon is long and risk aversion is high, even the results of “SGBM-LT” are not highly satisfactory. In that case, we prefer to solve the dynamic portfolio management problem by the COS portfolio management method. Regarding the simulation-based methods, “SGBM” and “SGBM-LT” are superior to “BGSS” and “vBB”, since their corresponding CERs have larger means and smaller standard errors.

Different from the findings in van Binsbergen and Brandt (2007b) that value function iteration also results in low certainty equivalent rates here. Table 3 shows that when using “SGBM” or “SGBM-LT”, we can also get satisfactory results by the value function iteration in most test cases. Portfolio weight iteration is significantly better than value function iteration when the risk aversion is large and the investment horizon is long.

## 5.4 Influence of Varying Initial State

We consider three different initial values,  $d_0$ , of the state variable. Each value corresponds to the  $p$ -th quantile of the unconditional distribution of  $d$ , where  $p$  takes values 25, 50 and 75.

Figure 2 shows that, for any initial state, “SGBM-LT” performs better than the other three simulation-based algorithms. The intermediate asset allocations generated by “SGBM-LT” are most similar to the optimal ones. At the initial recursion steps, “vBB” also generates similar asset allocations. However, as the backward recursion progresses, the uncertainty in the “vBB” estimates grows and hence the accuracy of “vBB” gets worse.

**Table 3** Mean and the standard derivations of the CER from 100 runs, comparing “SGBM” and “SGBM-LT” with the portfolio weight and the value function iteration; different investment horizons and risk aversion parameters

| Annualized certainty equivalent rate (%) |      |        |      |        |         |        |      |        |      |
|--|------|--------|------|--------|---------|--------|------|--------|------|
| $\gamma$                                 | SGBM |        |      |        | SGBM-LT |        |      |        | COS  |
|  | PWI  | (SE)   | VFI  | (SE)   | PWI     | (SE)   | VFI  | (SE)   |      |
| <i>T</i> = 40                            |      |        |      |        |         |        |      |        |      |
| 5  | 8.55 | (0.02) | 8.53 | (0.02) | 8.56    | (0.02) | 8.53 | (0.02) | 8.53 |
| 10                                       | 7.67 | (0.02) | 7.61 | (0.03) | 7.77    | (0.03) | 7.70 | (0.04) | 7.74 |
| 15                                       | 7.17 | (0.02) | 7.13 | (0.02) | 7.26    | (0.02) | 7.18 | (0.16) | 7.27 |
| 20                                       | 6.90 | (0.01) | 6.86 | (0.08) | 6.97    | (0.02) | 6.81 | (0.39) | 6.98 |
| <i>T</i> = 80                            |      |        |      |        |         |        |      |        |      |
| 5  | 8.96 | (0.01) | 8.94 | (0.02) | 8.97    | (0.01) | 8.94 | (0.02) | 8.94 |
| 10                                       | 8.08 | (0.02) | 8.02 | (0.02) | 8.28    | (0.02) | 8.18 | (0.15) | 8.29 |
| 15                                       | 7.47 | (0.01) | 7.42 | (0.05) | 7.65    | (0.02) | 7.55 | (0.11) | 7.83 |
| 20                                       | 7.13 | (0.01) | 7.09 | (0.04) | 7.28    | (0.01) | 7.16 | (0.17) | 7.49 |

The COS method serves as the reference

In any case, “SGBM” and “SGBM-LT” yield estimates with low uncertainties. Moreover, we see that “SGBM-VFI” and “SGBM-LT-VFI”, in which the value function iteration is considered in the recursion step, respectively, generate very similar results to those of “SGBM” and “SGBM-LT”. These are advantages of the new method to calculate conditional expectations.

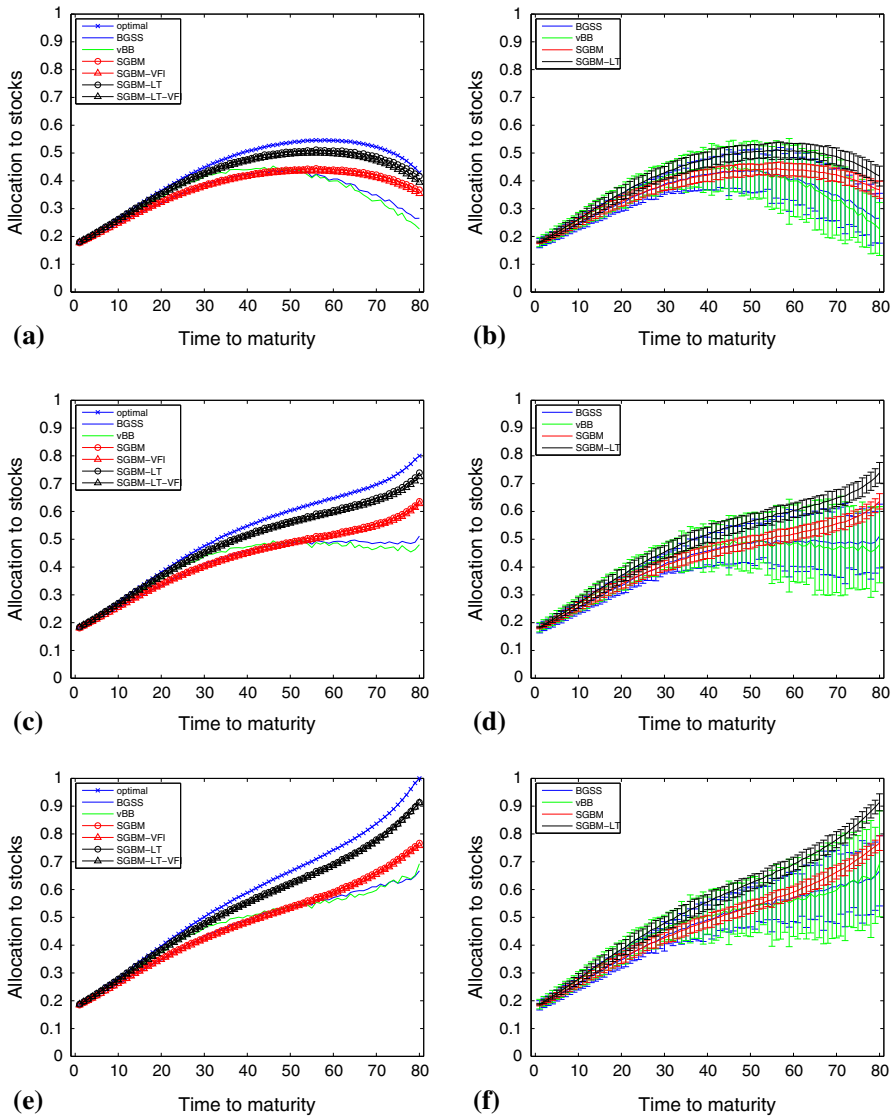
### 5.5 Influence of Varying Model Uncertainty

If we consider higher model uncertainty in the 1D-VAR model, the aforementioned methods perform differently. The model uncertainty can be modified by introducing a multiplier  $M^2$  to the original covariance matrix  $\Sigma_\epsilon$  of the white noise vector  $[\epsilon_{t+1}^r, \epsilon_{t+1}^d]'$ , so that the covariance matrix of the error term will be:

$$\Sigma_\epsilon^M = M^2 \cdot \Sigma_\epsilon.$$

In this test, with a fixed risk aversion parameter  $\gamma = 10$ , we change the multiplier  $M$  and the investment horizon and report the certainty equivalent rates corresponding to the different algorithms.

As shown in Table 4, when the model uncertainty increases, “vBB” is the most impacted algorithm. “BGSS” performs somewhat better than “vBB” but worse than “SGBM” and “SGBM-LT” as the corresponding certainty equivalent rate is smaller and with higher uncertainty. “SGBM-LT” outperforms “SGBM”. The differences are obvious when the model uncertainty is high and the investment horizon is long. The “SGBM-LT” values in the table are obtained with sample size  $2^{14}$ . In any case, “COS” yields the reference results, which are verified by using “SGBM-LT” with a large



**Fig. 2** Comparison of simulation-based algorithms for estimating the optimal intermediate asset allocations for different initial states. At each time step, the average asset allocations is computed. For the simulation-based algorithms, the mean and the standard deviation of the average asset allocations are generated from 100 runs. The optimal values are generated by the COS method, **a** Mean of the average asset allocations,  $d_0 = -3.8551$ , **b** Confidence interval of the average asset allocations,  $d_0 = -3.8551$ , **c** Mean of the average asset allocations,  $d_0 = -3.6905$ , **d** Confidence interval of the average asset allocations,  $d_0 = -3.6905$ , **e** Mean of the average asset allocations,  $d_0 = -3.5258$ , **f** Confidence interval of the average asset allocations,  $d_0 = -3.5258$

**Table 4** Comparing four methods with various model uncertainties.

| Annualized certainty equivalent rate (%) |      |        |       |        |      |        |         |        |      |
|--|------|--------|-------|--------|------|--------|---------|--------|------|
| <i>M</i>                                 | BGSS | (SE)   | vBB   | (SE)   | SGBM | (SE)   | SGBM-LT | (SE)   | COS  |
| <i>T</i> = 10                            |      |        |       |        |      |        |         |        |      |
| 1  | 6.64 | (0.03) | 6.64  | (0.03) | 6.65 | (0.03) | 6.65    | (0.03) | 6.64 |
| 2  | 6.65 | (0.03) | 6.65  | (0.03) | 6.66 | (0.03) | 6.66    | (0.03) | 6.65 |
| 3  | 6.79 | (0.38) | 6.59  | (0.82) | 6.84 | (0.05) | 6.86    | (0.03) | 6.85 |
| 4  | 6.98 | (1.05) | 0.11  | (6.46) | 7.11 | (0.04) | 7.14    | (0.04) | 7.13 |
| <i>T</i> = 20                            |      |        |       |        |      |        |         |        |      |
| 1  | 7.06 | (0.03) | 7.07  | (0.04) | 7.07 | (0.03) | 7.10    | (0.03) | 7.06 |
| 2  | 7.06 | (0.03) | 7.01  | (0.06) | 7.06 | (0.03) | 7.12    | (0.04) | 7.07 |
| 3  | 7.21 | (0.31) | 6.84  | (0.34) | 7.27 | (0.06) | 7.40    | (0.05) | 7.34 |
| 4  | 7.20 | (0.99) | -0.23 | (5.00) | 7.54 | (0.11) | 7.78    | (0.07) | 7.72 |

The risk aversion parameter is fixed as 10. For different investment horizon *T* (quarters) and multiplier *M* to the model uncertainty, the table reports the mean and the standard derivation of the certainty equivalent rate from 100 runs. The results from the COS method serve as the reference values

sample size  $2^{18}$ . In that case, we find, for example, the certainty equivalent rate of “SGBM-LT” has mean value 7.71 and standard error 0.02 when  $T = 20$  and  $M = 4$ .

### 5.6 Errors of the Four Simulation-Based Methods

In this subsection, we would like to briefly summarize the errors encountered within the methods analyzed. If we do not consider errors in the simulation part, the errors of the four simulation-based methods, “vBB”, “BGSS”, “SGBM” and “SGBM-LT”, can be subdivided into three categories:

- *approximation error* which occurs when we approximate the true value functions by the Taylor series expansion.
- *projection error* which occurs when we use low-order polynomials to approximate the conditional expectations of the value functions or of the approximated value functions.
- *regression bias* which occurs when we use cross-path regression to approximate the conditional expectations.

The *approximation error* does not occur when Taylor series expansions are not involved, for example, in “vBB”. However, as we have seen in the numerical tests, “BGSS” and “SGBM” suffer from this source of error in a similar fashion, while “SGBM-LT” appears to suffer less.

The *projection error* is the main source of error in “vBB”, where low-order polynomials are implemented to approximate the value functions, which may be high-order functions when the risk aversion is high, see Eq. (12). For “BGSS”, “SGBM” and “SGBM-LT”, this is generally not a problem since the object functions, as in Eq. (9), are at most of fourth-order.

**Table 5** Errors of the four simulation-based methods

|                     | vBB  | BGSS | SGBM | SGBM-LT |
|---------------------|------|------|------|---------|
| Approximation error | –    | High | High | Low     |
| Projection error    | High | Low  | Low  | Low     |
| Regression bias     | High | High | Low  | Low     |

The *regression bias*, which has been discussed in Cong and Oosterlee (2015), can be controlled effectively by bundling. The regression bias is high in “vBB” and “BGSS” but relatively low in “SGBM” and “SGBM-LT”, that benefit from their bundling technique.

A general description of the error components of the four simulation-based methods is listed in Table 5. “SGBM-LT” exhibits a highly satisfactory performance in our tests, since it has relatively small-sized errors in all three aspects. We expect however that when the risk aversion parameter is high and the model volatility is large, even “SGBM-LT” may fail to converge in some cases. In those cases, we propose either to use a large number of paths in the simulation together with more bundles or to implement a variance reduction technique.

## 6 Conclusion

In this paper, we enhance a popular dynamic portfolio management algorithm, the BGSS algorithm, in two aspects. First, for the computation of the conditional expectations appearing, we replace the standard regression method by the techniques from the SCBM, so that the variances of the approximated asset allocations and the corresponding certainty equivalent rates can be reduced. Then, a log Taylor expansion, based on a nonlinear decomposition, is employed in our algorithm. This expansion gives rise to improved results compared to the original ones when approximating the utility function. The resulting SGBM based portfolio management algorithm results in a lower biased approximation of the optimal asset allocations.

Based on the COS method and the grid-searching technique, we have developed the COS portfolio management method for generating reference values, which are quite comparable to the reference values and further serve as the “optimal” solutions in our numerical tests.

In our tests, combining SGBM and the log Taylor expansion yields superior results to those of other simulation-based algorithms. In all testing cases, “SGBM-LT” shows the higher certainty equivalent rates. When we merely consider introducing the SGBM components in the regression step, the benefits are obvious: the value function iteration and the portfolio weight iteration associated to both “SGBM” and “SGBM-LT” generate quite similar results, which indicate that the approximation errors at each recursion step are small.

Our simulation- and regression-based algorithm “SGBM-LT” can be generalized to higher-dimensional dynamic portfolio management problems. Besides, since our algorithm is robust even in scenarios with high volatility dynamics, it is also possible to focus on models with more complicated dynamics, for example, models with jump



components or other time series models. In those cases, we may need some effective bundling technique as proposed in Jain and Oosterlee (2015) and Cong and Oosterlee (2015) but in each local domain we may still use low-order polynomials as the basis functions. This helps to retain the robustness of our algorithm. In this paper, our benchmark method is only employed for the case of one risky asset. It can be extended at least to solving portfolio management problems with two or three risky assets. For all algorithms, it is promising to adopt parallel computation.

One potential future research direction is to combine the grid-searching approach with SGBM. This may be useful for utility-based optimization problems, where the utility function cannot be approximated accurately by its Taylor expansion. In that case, an adaptive grid-search combined with SGBM may constitute a generic solution.

**Open Access** This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

## References

- Barberis, N. (2000). Investing for the long run when returns are predictable. *The Journal of Finance*, 55(1), 225–264.
- Bellman, R. (1957). *Dynamic Programming*. Princeton: Princeton University Press.
- Brandt, M. W., Goyal, A., Santa-Clara, P., & Stroud, J. R. (2005). A simulation approach to dynamic portfolio choice with an application to learning about return predictability. *Review of Financial Studies*, 18(3), 831–873.
- Cochrane, J. H. (1989). The sensitivity of tests of the intertemporal allocation of consumption to near-rational alternatives. *Economic Review*, 79(3), 319–337.
- Cong, F., & Oosterlee, C. W. (2015). Pricing bermudan options under merton jump-diffusion asset dynamics. *International Journal of Computer Mathematics*, 92(12), 2406–2432.
- Cong, F., & Oosterlee, C. W. (2016). Multi-period mean-variance portfolio optimization based on Monte-Carlo simulation. *Journal of Economic Dynamics and Control*, 64(1), 23–38.
- Fang, F., & Oosterlee, C. W. (2008). A novel pricing method for European options based on Fourier-cosine series expansions. *SIAM Journal on Scientific Computing*, 31(2), 826–848.
- Fang, F., & Oosterlee, C. W. (2009). Pricing early-exercise and discrete barrier options by fourier-cosine series expansions. *Numerische Mathematik*, 114(1), 27–62.
- Garlappi, L., & Skoulakis, G. (2009). Numerical solutions to dynamic portfolio problems: The case for value function iteration using taylor approximation. *Computational Economics*, 33(2), 193–207.
- Garlappi, L., & Skoulakis, G. (2011). Taylor series approximations to expected utility and optimal portfolio choice. *Mathematics and Financial Economics*, 5(2), 121–156.
- Jain, S., & Oosterlee, C. W. (2015). The stochastic grid bundling method: Efficient pricing of Bermudan options and their Greeks. *Applied Mathematics and Computation*, 269(1), 412–431.
- Jondeau, E., & Rockinger, M. (2006). Optimal portfolio allocation under higher moments. *European Financial Management*, 12(1), 29–55.
- Longstaff, F. A., & Schwartz, E. S. (2001). Valuing American options by simulation: A simple least-squares approach. *Review of Financial Studies*, 14(1), 113–147.
- Merton, R. C. (1969). Lifetime portfolio selection under uncertainty: The continuous-time case. *The Review of Economics and Statistics*, 51, 247–257.
- Ruijter, M. J., & Oosterlee, C. W. (2012). Two-dimensional Fourier cosine series expansion method for pricing financial options. *SIAM Journal on Scientific Computing*, 34(5), B642–B671.
- Samuelson, P. A. (1969). Lifetime portfolio selection by dynamic stochastic programming. *The Review of Economics and Statistics*, 51, 239–246.

- Tsitsiklis, J. N., & Van Roy, B. (2001). Regression methods for pricing complex American-style options. *IEEE Transactions on Neural Networks*, 12(4), 694–703.
- van Binsbergen, J. H. & Brandt, M. W. (2007a). Optimal asset allocation in asset liability management. Technical report, National Bureau of Economic Research.
- van Binsbergen, J. H., & Brandt, M. W. (2007b). Solving dynamic portfolio choice problems by recursing on optimized portfolio weights or on the value function? *Computational Economics*, 29(3–4), 355–367.