Challenges in Financial Computing*

Martin B. Haugh and Andrew W. Lo†

This Draft: March 18, 2001

Abstract

One of the fastest growing areas of scientific computing is in the financial industry. Many of the most basic problems in financial analysis are still unsolved, and are surprisingly resilient to the onslaught of legions of talented researchers from many diverse disciplines. In this article, we hope to give readers a sense of these challenges by describing a relatively simple problem that all investors face—managing a portfolio of financial securities over time to optimize a particular objective function—and showing how complex such a problem can become as real-world considerations such as taxes, preferences, and portfolio constraints are incorporated into its formulation.

*This research was partially supported by the MIT Laboratory for Financial Engineering.

†MIT Sloan School of Management and Operations Research Center. Corresponding author: Andrew W. Lo, 50 Memorial Drive, E52-432, Cambridge, MA 02142–1347, (617) 253–0920 (voice), (617) 258–5727 (fax), alo@mit.edu (email).

Contents

1 Introduction

One of the fastest growing areas of scientific computing is in the financial industry. Two decades ago, terms such as "financial engineering", "computational finance", and "financial mathematics" did not exist in common usage, yet today these areas are regarded as distinct and enormously popular academic disciplines, with their own journals, conferences, and professional societies.

One explanation for the remarkable growth in this area and the impressive array of mathematicians, computer scientists, physicists, and economists that have been drawn to quantitative finance is the formidable intellectual challenges that are intrinsic to financial markets. Many of the most basic problems in financial analysis are still unsolved, and are surprisingly resilient to the onslaught of legions of researchers from diverse disciplines.

In this article, we hope to give readers a sense of these challenges by describing a relatively simple problem that all investors face—managing a portfolio of financial securities over time to optimize a particular objective function—and showing how complicated such a problem can become as real-world considerations are incorporated into its formulation. We present the basic dynamic portfolio optimization problem in Section 2, and then consider three aspects of the problem in Sections 3–5: taxes, investor's preferences, and portfolio constraints.[1] We conclude in Section 6 by reviewing one of the most promising class of methods for solving these types of problems.

2 The Portfolio Optimization Problem

Portfolio optimization problems are among the most well-studied problems in modern finance,[2] yet they continue to occupy the attention of financial academics and industry professionals, both because of their practical relevance and their computational intractabilities. The basic dynamic portfolio optimization problem consists of an individual investor's decisions for allocating wealth among various expenditures and investment opportunities over time so as to maximize some objective function—typically the investor's expected lifetime

[1] These three issues are by no means exhaustive, but are merely illustrative examples of the kinds of challenges faced by financial engineers today. See Haliassos and Michaelides (2000) and Haugh and Lo (2001) for examples of other computational issues in portfolio optimization.

[2] See for example Merton (1990).

utility—given the prices and price dynamics of goods and financial securities he purchases, and any constraints such as tax liabilities, loan repayment provisions, income payments, and other cash inflows and outflows that determine the investor's overall budget.

For expositional clarity, we start with a simple framework in which there are only two assets available to the investor: a bond that yields a riskless rate of return and a stock that yields a random return of either 10% or -10% with probability p and $1-p$, respectively. Denote by B_t and S_t the prices of the bond and stock at date t, respectively. Without loss of generality and for notational simplicity, we assume that $S_0 = \$1$ and the riskless rate of interest is 0% hence $B_t = 1$ for all $t \geq 0$. Finally, suppose that the investor's horizon spans only three dates, $t = 0, 1,$ and 2, so that the possible paths for the stock price process S_t are given in Figure 1. Of course, in practice, an investor has many assets to choose from over many dates and where the price of each asset can take on many values. But for illustrative purposes, this simpler specification is ideal because it contains all the essential features of the dynamic portfolio optimization problem in a very basic setting. Nevertheless, even in this simple framework, it will become apparent that practical considerations such as taxes, investor preferences, and portfolio constraints can create surprisingly difficult computational challenges.

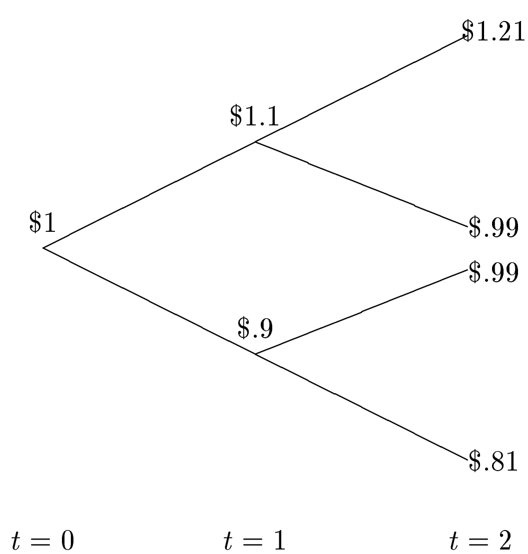

Figure 1: Stock Price Evolution

Let C_t denote the consumption expenditures of the investor at date t and let W_t denote the investor's wealth just *prior* to date-t consumption. We assume that the investor has a lifetime utility function $U(C_0, C_1, C_2)$ defined over each consumption path $\{C_0, C_1, C_2\}$ that summarizes how much he values the entire path of consumption expenditures. Then, absent market frictions and assuming that the investor's utility function is time-additive and time-homogeneous, i.e.,

$$U(C_0, C_1, C_2) \;=\; u(C_0) \;+\; u(C_1) \;+\; u(C_2) \tag{1}$$

the investor's dynamic portfolio optimization problem at $t=0$ is given by:

$$V_0(W_0) \;=\; \underset{C_0, C_1, C_2}{\mathrm{Max}} \; \mathrm{E}_0[u(C_0) + u(C_1) + u(C_2)] \tag{2}$$

subject to

$$W_t \;-\; C_t \;=\; x_t S_t + y_t B_t \quad , \quad t = 0, 1, 2 \tag{3a}$$

$$W_{t+1} \;=\; x_t S_{t+1} + y_t B_{t+1} \quad , \quad t = 0, 1 \tag{3b}$$

$$C_t \;\geq\; 0 \quad , \quad t = 0, 1, 2 \tag{3c}$$

$$x_t,\; y_t \;\in\; \mathcal{Z}^+ \quad , \quad t = 0, 1, 2 \tag{3d}$$

$$x_2 \;=\; y_2 \;=\; 0 \tag{3e}$$

where $\mathcal{Z}^+$ denotes the non-negative integers, and x_t and y_t are the number of shares of stocks and bonds, respectively, that the investor holds in his portfolio immediately after date t.[3] The requirement that x_t and y_t are non-negative means that borrowing and short sales are not allowed, a constraint that many investors face. That x_t and y_t are required to be non-negative integers simply reflects the fact that it is not possible to purchase a fractional number of stocks or bonds. The constraint (3b) states that W_{t+1} is equal to W_t multiplied by the gross return on the portfolio between dates t and $t+1$.

This problem is easily solved numerically using the standard technique of stochastic

[3] Of course x_t and y_t may depend only on the information available at time t, a restriction that we impose throughout this article.

dynamic programming.[4] In particular, since $V_2(W_2) = u(W_2)$, we can then compute $V_1(W_1)$ using the Bellman equation so that

$$V_1(W_1) \quad = \quad \underset{C_1}{\mathrm{Max}} \left\{ u(C_1) + \mathrm{E}_1[V_2(W_2)] \right\} \tag{4}$$

subject to the constraints in (3). An aspect of (4) that makes it particularly easy to solve is the fact that the value function $V_1(\cdot)$ depends on only one state variable, W_1. This enables us to solve (4) numerically without too many computations. Suppose, for example, that $W_0 = \$1{,}000$. Then, since $x_t,\ y_t \in \mathcal{Z}^+$, there are only 1,100 possible values that W_1 can take, so the right side of (4) must be evaluated for only these 1,100 values. In contrast, when there are market frictions or when the investor has a more complex utility function, we will see how the computational requirements increase dramatically, reflecting Bellman's "curse of dimensionality".

3 Taxes

Most seasoned investors are painfully aware of the substantial impact that taxes can have on the performance of their investment portfolio, hence taxes play a major role in most dynamic portfolio optimization problems.[5] To see how taxes can increase the computational complexity of such problems, let trading profits in the stock be subject to a capital gains tax in the simple model of Section 2. Because this model has only two future periods, we do not distinguish between short-term and long-term capital gains, and for expositional simplicity we also assume that capital losses from one period cannot be used to offset gains from a later period. Even though these simplifying assumptions do make the problem easier to solve, nevertheless, it will be apparent that the computations are considerably more involved than in the no-tax case.

To solve the dynamic portfolio optimization problem with taxes, we use dynamic programming as before. However, the value function is now no longer only a function of wealth, but also depends on past stock prices and the number of shares of stock purchased at each of

[4] See, for example, Bertsekas (1995).

[5] See, for example, Bertsimas, Lo and Mourtzinou (1998), Dybvig and Koo (1996), and Constantinides (1983).

those prices. In other words, the value function is now path-dependent. If we use $N_{s,t}$ to denote the number of shares of stock that was purchased at date $s \leq t$ and still in the investor's portfolio immediately after trading at date t, then the portfolio optimization problem may be expressed as:

$$V_0(W_0) \;=\; \underset{C_0,C_1,C_2}{\mathrm{Max}} \;\; \mathrm{E}_0\,[\,u(C_0) + u(C_1) + u(C_2)\,] \tag{5}$$

subject to

$$\begin{aligned} W_t \;-\; C_t \;-\; \mathrm{Max}\left\{\,0\,,\,\sum_{s=0}^{t-1} \tau(S_t - S_s)(N_{s,t-1} - N_{s,t})\,\right\} \;&=\; \\ & x_t S_t \;+\; y_t B_t \quad , \quad t = 0,1,2 \end{aligned} \tag{6a}$$

$$W_{t+1} \;=\; x_t S_{t+1} \;+\; y_t B_{t+1} \quad , \quad t = 0,1 \tag{6b}$$

$$x_t \;=\; \sum_{s=0}^{t} N_{s,t} \quad , \quad t = 0,1,2 \tag{6c}$$

$$C_t \;\geq\; 0 \quad , \quad t = 0,1,2 \tag{6d}$$

$$N_{0,0} \;\geq\; N_{0,1} \;\geq\; N_{0,2} \;\geq\; 0 \tag{6e}$$

$$N_{1,1} \;\geq\; N_{1,2} \;\geq\; 0 \tag{6f}$$

$$x_t\,,\,y_t \;\in\; \mathcal{Z}^{+} \quad , \quad t = 0,1,2 \tag{6g}$$

$$x_2 \;=\; y_2 \;=\; 0\;. \tag{6h}$$

where τ is the capital gains tax rate. When $t=2$, the value function depends on $(W_2, S_1, S_2, N_{0,1}, N_{1,1})$ and we obtain the relation:

$$V_2(W_2, S_1, S_2, N_{0,1}, N_{1,1}) \;=\; u\left(W_2 - \mathrm{Max}\{0, \sum_{s=0}^{1} \tau(S_2 - S_s)N_{s,1}\}\right) \tag{7}$$

so that date-2 consumption is simply W_2 less any capital gains taxes that must be paid. At $t=1$, the value function depends on $(W_1, S_1, N_{0,0})$ and we can write the Bellman equation as

$$V_1(W_1, S_1, N_{0,0}) \;=\; \underset{C_1}{\mathrm{Max}} \left\{\, u(C_1) \;+\; \mathrm{E}_1[V_2(W_2, S_1, S_2, N_{0,1}, N_{1,1})] \,\right\}. \tag{8}$$

If we compare (8) with (4), it is apparent that the presence of taxes has made the dynamic portfolio optimization problem considerably more difficult. Specifically, in solving (4) numerically, $V_1(\cdot)$ is computed for only 1,100 possible values of W_1. In contrast, solving (8) numerically requires the evaluation of $V_1(\cdot)$ for all possible combinations of $\{W_1, S_1, N_{0,0}\}$, of which there are 1,001,000![6] Even in a simple two-period two-asset model, the portfolio optimization problem with taxes becomes considerably more complex. Indeed, in the T-period N-asset case, it is easy to see that by date T, there are $\mathbf{O}(NT)$ state variables, and if each state variable can take on m distinct values, then there will be $\mathbf{O}(m^{NT})$ possible states at date T. For an investor with a 20-year horizon, an annual trading interval, and a choice of 25 assets at the start, and assuming that the state variables take on only 4 distinct values at the end of the horizon,[7] the number of possible states at the end will be of the order 10^{301}.

4 Preferences

Another important aspect of portfolio optimization problems is the objective function that represents the investor's preferences. Traditionally, these preferences have been represented by time-additive time-homogeneous utility functions, which yields important computational advantages because it implies that the value function at date t does not depend on the investor's consumption choices prior to date t.

Unfortunately, the assumptions of time-additivity and time-homogeneity seem to be inconsistent with the empirical evidence on the consumption and portfolio choices of investors. For example, it is well known that individuals tend to grow accustomed to their level of consumption over a period of time, implying that preferences depend not only on today's con-

[6] This can be verified by noting the one-to-one correspondence between $\{W_1, S_1, N_{0,0}\}$ and $\{C_0, S_1, N_{0,0}\}$, and counting the possible combinations of $\{C_0, S_1, N_{0,0}\}$. Assuming, as before, that $W_0 = \$1{,}000$, we see that there are 1,001 possible choices for C_0. If $C_0 = i$, then there are $1{,}000 - i$ possible values for $N_{0,0}$. For each combination of $(C_0, N_{0,0})$, there are 2 possible values of S_1. This means that in total, there are

$$2 \sum_{i=0}^{1,000} (1{,}000 - i) \quad = \quad 1{,}001{,}000$$

combinations of $\{W_1, S_1, N_{0,0}\}$.

[7] The number of distinct values that a state variable can take on depends, of course, on the precise nature of the state variable. For example, for a binomial state variable that is not "recombining", the number of distinct values it can take on is 2^T; if it is recombining, this is reduced to $T+1$. We use 4 only for illustrative purposes; in most practical applications, the number is considerably larger.

sumption level but also on levels of past consumption.[8] Commonly known as "habit formation", such preferences imply that the value function $V_t(\cdot)$ is a function of $(C_0, C_1, \ldots, C_{t-1})$ in addition to any other relevant state variables. As in the case with taxes, these problems quickly become intractable as the number of time periods increases. Of course, there do exist a few highly parametrized models of habit formation in which closed-form solutions are available,[9] but in general these models must be solved numerically as in Heaton (1995). There are many other empirical regularities of investors' preferences that can induce path dependence in the value function, and for each of these cases, the computational demands quickly become intractable.

5 Portfolio Constraints

When constraints are imposed on a portfolio optimization problem, their impact on the computational complexity of the problem is not obvious. On the one hand, some unconstrained problems that admit closed-form solutions fail to do so once constraints are added. In practice, however, closed-form solutions are rarely available for realistic portfolio optimization problems, with or without portfolio constraints. Such problems must be solved numerically, in which case, imposing constraints can sometimes reduce the number of computations since they limit the feasible region over which the value function must be evaluated. An example of this is the impact of the constraints in (3) on the basic portfolio optimization problem described in Section 2. In that case, we imposed the constraint that x_t and y_t are non-negative integers, eliminating the possibility of borrowing or shortselling. This implies that only a finite number of values for W_1 are possible, and as a result, the number of computations needed to evaluate $V_1(W_1)$ is greatly reduced.

On the other hand, in some cases constraints can greatly increase the number of computations, despite the fact that they limit the feasible set. This typically occurs when the constraints increase the dimensionality of the problem. For example, in the basic portfolio optimization problem of Section 2, consider imposing the additional constraint that the cumulative number of shares transacted—both purchased *and* sold—up to date t is bounded

[8] See Kahneman, Slovic, and Tversky (1982) for other examples of preferences that account for the foibles of individual behavior.

[9] For example, see Constantinides (1990).

by some function, $f(t)$.[10] In practice, these types of constraints are often imposed on investment funds so as to reduce transactions costs and the risk of "churning". When such a constraint is imposed, the value function is no longer a function of only W_t, but also of the cumulative number of shares transacted up to date t. By creating path dependence in the value function, constraints can substantially increase the computational complexity of even the simplest portfolio optimization problems.

6 Possible Solution Techniques

As discussed in Section 2, the most natural technique for solving dynamic portfolio optimization problems is stochastic dynamic programming. However, this approach is often compromised by several factors such as the curse of dimensionality when too many state variables are involved, as in Sections 3–5. In general, practical considerations such as taxes, transactions costs, indivisibilities and integer constraints, non-time-additive utility functions, and other institutional features of financial markets tend to create path dependencies in portfolio optimization problems, which increases the number of state variables in the value function. Such problems are very difficult to solve in all but the simplest cases, with computational demands that become prohibitive as the number of time periods and assets increase.

In this section, we briefly outline an alternative that may produce good approximate solutions to otherwise intractable portfolio optimization problems. This approach, called "approximate dynamic programming", "neuro-dynamic programming", or "reinforcement learning", has had much success recently in solving challenging dynamic optimization problems in several contexts, and is described in more detail in Bertsekas and Tsitsiklis (1996). The method has already been applied successfully in one financial context by Longstaff and Schwartz (2001), who use the technique to price high-dimensional American options. Although there are many different algorithms that may be categorized as "approximate dynamic programming", we confine our attention in this section to just one such algorithm: approximate value iteration.

Suppose the optimal value function at date t of a T-period dynamic optimization problem is given by $V_t(X_t^1, \ldots, X_t^n)$ where $(X_t^1, \ldots, X_t^n)$ are state variables. We assume that because

[10] If 500 shares were purchased at $t=0$ and 200 shares were sold at $t=1$, the cumulative number of shares transacted as of date $t=1$ is 700.

of the computational intractabilities, it is impossible to determine $V_t(\cdot)$ exactly. Therefore, we define a parametrized class of functions

$$\left\{ \widetilde{V}_t(X_t^1, \ldots, X_t^n; \beta_t) \;:\; \beta_t \in \mathcal{R}^p \right\} \tag{9}$$

which we call the *approximation architecture*. We then select our estimate $\widetilde{V}_t(X_t^1, \ldots, X_t^n; \hat{\beta}_t)$ of V_t from this class of functions by selecting $\hat{\beta}_t$.

Now suppose that we have applied the backward recursion of the Bellman equation to obtain estimators $\widetilde{V}_{t+1}, \ldots, \widetilde{V}_T$. Then we can use an approximate Bellman equation to compute an estimator of the value function at time t so that

$$\widehat{V}_t(X_t^1, \ldots, X_t^n) \;\; = \;\; \mathrm{Max}\;\; \mathrm{E}_t[\widetilde{V}_{t+1}(X_{t+1}^1, \ldots, X_{t+1}^n; \hat{\beta}_{t+1})] \tag{10}$$

where the maximization is with respect to the decision variables, which have been suppressed for notational simplicity. Computing $\widehat{V}_t(\cdot)$ will often entail extensive Monte Carlo simulations since it is generally not possible to compute expectations over a high-dimensional space. In these cases, we estimate $\widehat{V}_t(\cdot)$ at a fixed number of "training points" $(P_1, \ldots, P_m)$, where each $P_i \in \mathcal{R}^n$ represents a possible realization of the state vector at date t. Once we have estimated $\widehat{V}_t(P_i)$ for $i = 1, \ldots, m$, we obtain $\widetilde{V}_t$ by solving the following least squares problem:

$$\widehat{\beta}_t \;\; = \;\; \arg \underset{\beta_t}{\mathrm{Min}} \sum_{i=1}^{m} \left(\widehat{V}_t(P_i^t) - \widetilde{V}_t(P_i^t; \beta_t) \right)^2 . \tag{11}$$

With $\widetilde{V}_t$ now determined, we then proceed to compute $\widetilde{V}_{t-1}$ in a similar fashion, and continue in this manner of approximate value iteration until we have found $\widetilde{V}_0$.

Once an approximate dynamic programming algorithm has been implemented, an estimator $\{\widetilde{V}_t\}$ of the value function is obtained. The natural question that follows is whether or not this estimator is "good". While there are some theoretical results that provide a partial answer to this question,[11] it is often very difficult in practice to determine the accuracy of an approximate solution to a particular problem. One possibility is to attempt to derive lower and upper bounds on the true value function, V_t. Deriving a lower bound is typically

[11]See, for example, Bertsekas and Tsitsiklis (1996).

straightforward—the sequence $\{\widetilde{V}_t : t = 1, \ldots, T\}$ defines a feasible trading strategy and therefore the value of this strategy, which can be estimated via simulation, is a lower bound for V_0. However, deriving an upper bound for V_0 is generally not so straightforward, but one possibility is to apply stochastic duality theory, which has already been studied extensively in the context of portfolio optimization.[12] In addition, Haugh and Kogan (2001) have successfully employed duality theory in conjunction with approximate dynamic programming to construct lower and upper bounds on the prices of American options. It is possible that a similar approach might also work for portfolio optimization problems, which we are investigating in ongoing research.

There are many other approximate dynamic programming solutions, including algorithms based on approximate policy iteration and on Q-learning.[13] These approaches generally share the common feature of resorting to function approximation and simulation techniques to deal with computational intractabilities. As computing power continues its remarkable growth, we believe that these techniques will become increasingly important in addressing many of the challenges of financial computing over the next few decades.

[12]See, in particular, Karatzas and Shreve (1998).

[13]See Bertsekas and Tsitsiklis (1996).

References

Bertsekas, D., 1995, *Dynamic Programming and Optimal Control*, Vol. I. Belmont, MA: Athena Scientific.

Bertsekas, D. and J. Tsitsiklis, 1996, *Neuro-Dynamic Programming*, Belmont, Massachusetts: Athena Scientific.

Bertsimas, D., A. W. Lo and G. Mourtzinou, 1998, "Tax-Aware Multiperiod Portfolio Optimization", unpublished working paper, MIT Sloan School of Management.

Constantinides, G. M., 1983, "Capital Market Equilibrium with Personal Tax", *Econometrica* 51, 611–636.

Constantinides, G. M., 1990, "Habit Formation: A Resolution of the Equity Premium Puzzle", *Journal of Political Economy* 98, 519–543.

Dybvig, P. H. and H. K. Koo, 1996, "Investment with Taxes", unpublished working paper, Olin School, Washington University in St Louis.

Haliassos, M. and A. Michaelides, 2000, "Calibration and Computation of Household Portfolio Models", forthcoming in L. Guiso, M. Haliassos and T. Japelli, eds., *Household Portfolios.* Cambridge, MA: MIT Press.

Haugh, M. and L. Kogan, 2001, "Pricing American Options: A Duality Approach", unpublished working paper, the Wharton School, University of Pennsylvania.

Haugh, M. and A. Lo, 2001, "Asset Allocation and Derivatives", *Quantitative Finance* 1, 42–75.

Heaton, J., 1995, "An Empirical Investigation of Asset Pricing with Temporally Dependent Preference Specifications", *Econometrica* 63, 681–717.

Kahneman, D., Slovic, P. and A. Tversky, 1982, *Judgment Under Uncertainty: Heuristics and Biases.* Cambridge, UK: Cambridge University Press.

Karatzas, I. and S. Shreve, 1998, *Methods of Mathematical Finance.* New York: Springer Verlag.

Longstaff, F. and E. Schwartz, "Valuing American Options By Simulation: A Simple Least Squares Approach", forthcoming in *Review of Financial Studies.*

Merton, R., 1990, *Continuous-Time Finance.* Oxford, UK:Basil Blackwell.