Available online at www.sciencedirect.com

SCIENCE DIRECT®

PERGAMON

Computers & Operations Research 31 (2004) 1–20

computers & operations research

www.elsevier.com/locate/dsw

Financial planning via multi-stage stochastic optimization

John M. Mulvey[a], Bala Shetty[b,*]

[a] Department of Operations Research and Financial Engineering, Princeton University, Princeton, NJ 08544, USA
[b] Department of Information and Operations Management, Texas A&M University, 322 Wehner Building, College Station, TX 77843-4217, USA

Received 1 February 2000; received in revised form 1 July 2001

Abstract

This paper describes a framework for modeling significant financial planning problems based on multi-stage optimization under uncertainty. Applications include risk management for institutions, banks, government entities, pension plans, and insurance companies. The approach also applies to individual investors who are interested in integrating investment choices with savings and borrowing strategies. A dynamic discrete-time structure addresses realistic financial issues. The resulting stochastic program is enormous by current computer standards, but it possesses a special structure that lends itself to parallel and distributed optimization algorithms. Interior-point methods are particularly attractive. Solving these stochastic programs presents a major challenge for the computational operations research and computer science community.

Scope and purpose

The globalization of financial markets and the introduction of complex products such as exotic derivatives have increased volatility and risks. Strides in computer and information technology has eliminated any delays between the occurrence of an event and the impact on the markets—within the home country and internationally. Thus there is a great need for an integrative approach to financial analysis and planning that encompasses the decisional environment as well as the stochastic elements in a dynamic fashion. The financial optimization model presented here incorporates several popular approaches to the problem of investment strategies, including stochastic programming and dynamic stochastic control.

Keywords: Financial optimization; Stochastic programming; Nonlinear programming

* Corresponding author. Tel.: +1-979-845-1616; fax: +1-979-845-5653.
E-mail address: b-shetty@tamu.edu (B. Shetty).

0305-0548/04/$ - see front matter © 2003 Elsevier Ltd. All rights reserved.
PII: S0305-0548(02)00141-7

0. Introduction

This paper presents a general framework for modeling financial planning problems via multi-stage stochastic programming. The framework addresses a wide spectrum of realistic issues in financial planning. It integrates investment strategies (also known as asset allocation strategies), liability decisions (e.g., borrowings) and savings strategies (or re-investment decisions) in an integrative fashion. There is great need for an integrative approach to financial analysis and planning [1–3]. The globalization of markets and the explosion in new forms of securities has greatly complicated the investment problem. The financial rewards for good decisions and penalties for bad decisions are enormous. Uncertainty plays a critical role in the process. Strides in computers and information technology has eliminated any delays between the occurrence of an event (such as central bank decisions regarding the direction of future interest rates) and the impact on the markets—within the home country and internationally. The recent wave of improper investment decisions using modern financial instruments such as derivatives and the futures market illustrates the penalties for poor risk management. It is widely believed that the losses of $1.2 billion at Orange County, $900 million at Barings, and $300 million at Kidder Peabody could have been averted by a more careful and integrated risk management approach.

Insurance companies pioneered some of what is known today as integrated risk management. Noteworthy applications include the Russell–Yasuda Kasai investment system for insurance companies [4], the Towers Perrin investment system for pension plans [5], the integrated simulation and optimization system for the Metropolitan Life Insurance Company [6], and the integrated product management system [7]. In each case, asset investment decisions are combined with liability choices and investor's wealth is maximized over time via multi-stage stochastic programming. Besides insurance companies, the framework discussed here may be used as a basis for assessing and managing risks by large institutional organizations, including banks, savings and loans, pension plans, university endowments, and government entities. Individual investors can also use the methodology for managing their financial affairs over time.

An integrated financial planning system consists of four main elements: a mathematical model, a solution algorithm, an automated information system, and an interface environment. In this article, we focus on the first two elements. While the modeling framework here is similar to those that appeared in [3,4,13], this article presents algorithmic and implementation issues that are new and innovative. In Section 1, we discuss a multi-stage stochastic programming model that has the ability to capture many factors involved in financial planning. It takes into account the decision-maker's risk preference, in addition to incorporating growth, budgetary, legal, and institutional policy restrictions. Section 1.1 presents the selection of an appropriate objective function and Section 1.2 describes various advantages of a multi-period framework. A critical issue involves the modeling of the stochastic parameters. We represent these parameters as a set of scenarios, generation of which is described in Section 1.3. Another key feature of the proposed framework is the integration of assets and liabilities, which is discussed in Section 1.4. In Section 2, we present some preliminary computational results for a six-stage investment problem. The investment problem is first solved using an enhancement to the direct solver LOQO [8]. The enhancement involves a tree dissection heuristic that significantly reduces the number of floating point operations. A parallel algorithm is then applied to our test problem using up to 128 processors. Despite the progress made by direct solvers, these solvers are not necessarily the best alternative when the number of scenarios grows very large. Consequently, in

Section 2.2, we explore a decomposition algorithm known as the diagonal approximation algorithm (DQA). Preliminary results for a parallel implementation of DQA are also discussed in this section. Section 3 presents financial planning via dynamic stochastic control. Certain decision rules are used to reduce the problem size. The resulting model is nonconvex requiring extensive search for a global solution. Conclusion and directions for future research are given in Section 4.

1. Multi-stage financial planning model

Various formulations for the multi-stage financial problem have appeared in the literature (e.g., [9–13]). Here we describe a generic financial planning problem as a multi-stage stochastic program. Many applications can be posed as special cases of this model. The problem is portrayed as a network graph as shown in Fig. 1. While some real-world issues are difficult to accommodate within the network context and must be handled as general linear constraints, the network provides a visual reference for the financial planning system.

The planning horizon consists of τ time periods represented by $T = \{0, 1, \ldots, \tau - 1\}$. The first period represents the current date. Period τ defines the date of the planning horizon; we focus on the investor's position at the beginning of period τ. Typically, it depicts a point at which the investor has some critical planning purpose, such as the repayment of a substantial liability. Decisions occur at the last instant of each time stage.

Asset investment categories are defined as set $A = \{1, 2, \ldots, I\}$ representing broad investment groupings such as stocks, bonds, real estate, or cash. The categories should track a well-defined market segment. Ideally, the co-movements between pairs of asset returns should be relatively low so that diversification can be done across the asset categories. There should be an index which tracks the market segment, for instance the S&P 500 index, the Russell 3000 index, or Morgan Stanley's international index. For convenience, the asset category should be available as a marketable security via index funds or a futures contract.

We model uncertainty using a large number of scenarios, each of which represents a single set of outcomes for all of the random coefficients over the entire planning period T [28]. Let S represent the set of scenarios that is a reasonable representation of the uncertain future. A scenario $s \in S$ then is a path through consecutive nodes of the scenario tree as shown Fig. 2 below. Developing this representative set continues to be a challenging research area and we discuss some of these challenges in a later section.

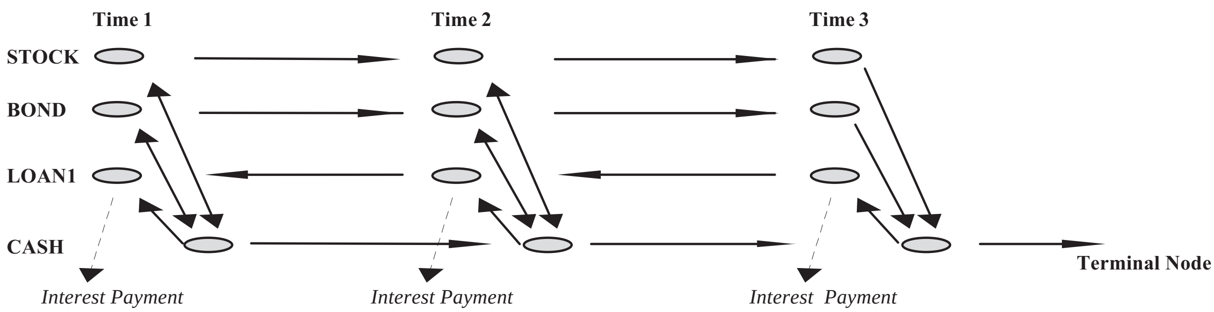

Fig. 1. Network model for financial planning.

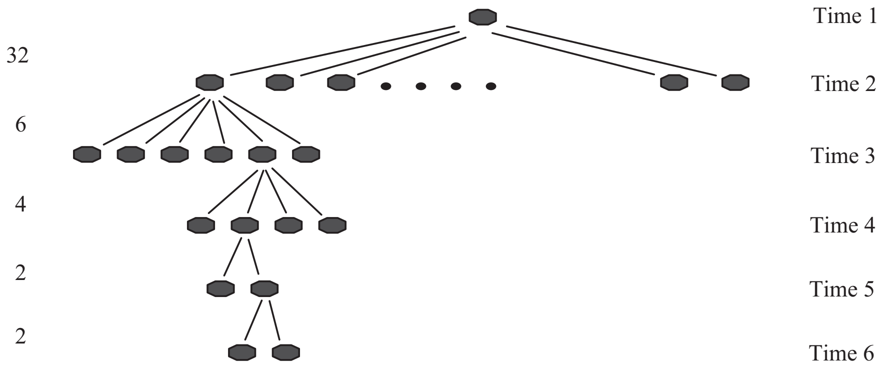

Fig. 2. Scenario tree.

In our approach, the primary decision variable $x_{i,t}^s$ denotes the amount of assets invested in category i at the beginning of time period t under scenario s. Units are consistent with the investor's home country. Thus, for instance, if an investor decides not to hedge currency risks, returns are always defined in terms of the investor's original currency. The basic model can be readily extended in order to deal with multi-currency variables, but model size grows as a consequence. The *x-vector* depicts the state of the system after the rebalancing decisions have been made in the previous period. At that time the investor's total assets are equal to

$$\sum_i x_{i,t}^s = a_t^s, \quad s \in S, \ t \in T. \tag{1}$$

The uncertain returns $r_{i,t}^s$ for the asset categories—asset i, time t—are projected by the stochastic modeling subsystem for scenario s. Each scenario is internally consistent. Thus $v_{i,t}^s$, the wealth accumulated at the end of the tth period before rebalancing in asset i, is given by

$$x_{i,t}^s(1 + r_{i,t}^s) = v_{i,t}^s \quad \forall i \in A, \ t \in T, \ s \in S. \tag{2}$$

Rebalancing decisions are rendered at the very end of each period. Purchases and sales of assets are accommodated by the variables $y_{i,t}^s$ and $z_{i,t}^s$ with transaction costs defined via the coefficients ξ_i assuming symmetry in the transaction costs. The flow balance constraint for each asset category and time period is defined as

$$x_{i,t}^s = v_{i,t-1}^s + y_{i,t-1}^s(1 - \xi_i) - z_{i,t-1}^s(1 + \xi_i) \quad \forall i \in A, \ t \in T, \ s \in S. \tag{3}$$

This equation restricts the cash flows at each period to be consistent. Each node in Fig. 1 corresponds to a flow balance equation of the type shown in (6) below. Dividends and interest are forthcoming simultaneously with the rebalancing decisions. Thus, the z variables consist of two parts corresponding to the involuntary cash outflow—dividend or interest, and a voluntary component for the cash flow—the amount actively sold (sales). The requisite equation is

$$z_{i,t}^s = d_{i,t}^s + u_{i,t}^s \quad \forall i \in A, \ t \in T, \ s \in S, \tag{4}$$

where $d_{i,t}^s$ represents the dividends and $u_{i,t}^s$ the amount of assets actively sold. The dividend equation is

$$d_{i,t}^s = x_{i,t}^s \psi_i^s, \tag{5}$$

where ψ_i^s indicates the dividend pay out percentage ratio for asset i under scenario s. The cash node at each period t also requires a flow balancing equation

$$c_t^s = g_{t-1}^s + \sum_i (u_{i,t-1}^s(1-\xi_i) + d_{i,t-1}^s) - \sum_i (y_{i,t}^s) - b_{t-1}^s(1-\delta_t^s) + c_{t-1}^s - e_{t-1}^s - f_{t-1}^s + b_t^s \quad \forall t \in T,\ s \in S \tag{6}$$

with four new decision variables: b_t^s is the amount of borrowing in each period t at the borrowing rate δ_t^s; e_{t-1}^s are the cash outflows; g_{t-1}^s are the cash inflows, and f_{t-1}^s is the paydown of principle in period $t-1$. Any of these decision variables may be dependent upon the state of the world represented by scenario s. For simplicity, assume that all borrowing is done on a single-period basis. Adding new decision variables for each category of multi-period borrowing avoids the assumption. Define initial wealth in asset $i \in A$ at the end of period 0 as $v_{i,0}$.

In the real world, investors restrict their investments in asset categories for a diversity of purposes such as company policy, legal and historical rules, and other considerations. These policy constraints may take any form. We limit their structure to the following set of linear restrictions:

$$B^s x^s = b^s \quad \forall s \in S, \tag{7}$$

where B is a matrix of coefficients that depend upon scenario s. For example, investors may set a lower limit—say 5%—on cash for liquidity considerations. Investors may wish to restrict their foreign exposure to 10–20% of their portfolio's value. These constraints are valuable on several counts. First, they increase the likelihood that the investment recommendations will be implemented. Adding constraints is the most direct way to approach the what-if analysis that is an inevitable byproduct of asset allocation studies. Investors must be convinced that the proposed decisions are sensible.

The variables and parameters in (1)–(7) have been indexed by s to indicate their dependence to scenario $s \in S$. These constraints decompose into subproblems one for each scenario, implying that we can anticipate a priori which scenario will materialize. For the model to be useful, however, we must include *nonanticipativity* conditions. These conditions require that groups of scenarios with identical values for the uncertain parameters up to a certain period must yield the same decisions up to that period. Mathematically, the conditions can be represented using equal flows on certain set of variables as

$$x_{i,t}^s = x_{i,t}^{s'} \tag{8}$$

for scenarios s and s' inheriting an identical past up to time t. The conditions stipulate that decision variables must be equal to each other as long as they have a common historical past until some time t in the planning horizon $\{0, 1, \ldots, \tau - 1\}$. While these constraints are extremely numerous, solution algorithms take advantage of their simple form—a pair of $+1$ and -1 for each row.

An alternative method for addressing nonanticipativity is to impose a set of control policies that do not depend upon knowing the future. We call this approach dynamic stochastic control (DSC). The well-known constant-mix and fixed-mix strategies are special cases of DSC [14]. As an example of the fixed-mix strategy, we could limit investment choices at each stage to a fixed proportion of wealth, which means purchases and sales may be required to bring the assets to the desired mix. Letting λ_i be the proportion of asset wealth invested in asset i, we define a set of constraints that prevent the model from using future information in rendering today's decisions. These constraints are shown as

$$\lambda_i = x^s_{i,t}/a^s_t \tag{9}$$

for asset i at time t. Constraint (9) forces a fixed investment of λ_i in asset i under any scenario s and time period t. Replacing each $x^s_{i,t}$ by $(a^s_t * \lambda_i)$ in (1)–(7) above greatly reduces the problem, resulting in a multi-stage financial planning model in decision variables λ_i for $i \in I$. Unfortunately, adding constraint (9) results in a nonconvex optimization problem that is harder to solve. Section 3 provides further details for solving the nonconvex problem. Once the optimal λ_i for each i are identified, the investor rebalances the portfolio to achieve the desired mix.

1.1. Objective functions

A fundamental issue in carrying out a financial modeling effort is to settle on the choice of an objective function and the underlying preference structure. In our basic model, the proposed objective function maximizes the investor's wealth at the beginning of period τ, subject to the pay out of intermediate cash outflows (liabilities) under each of the $s \in S$ scenarios. The investor's total wealth at the horizon τ under scenario s equals the following:

$$wealth^s_\tau = \sum_i x^s_{i,\tau} - PV(l^s_{\tau,f}) - b^s_t, \tag{10}$$

where $PV(l^s_{\tau,f})$ is the present value of the liability stream from period τ to some future period f and b^s_τ depicts the amount of loans outstanding at time period τ. This calculation may cause no particular difficulty. In many cases, however, the investor's liabilities are not readily marketable and must be projected along with the accompanying discount factors.

We extend the concept of wealth to encompass investment goals (or targets). First, the investor selects a set of goals for the accompanying time period; for instance, college tuition or a retirement annuity. Next, the scenario generation process is enhanced to compute the cash flows for achieving the goal as a function of the underlying economic factors such as inflation rate and interest rates. The wealth calculation is then modified to include investment goals within the liability framework. We call the modified result surplus wealth [15]. It provides an actuarially sound indicator for the investor's ability to meet a set of goals. The notion is well understood in the context of pension plans, but surplus wealth has rarely been applied to other situations due to computational and related limitations.

Several alternative objective functions are available for addressing the stochastic aspects of the problem. One possibility is to employ the classical mean-variance function

$$\text{Maximize } (1-\pi)Mean(W_\tau) - \pi\, Var(W_\tau), \tag{11}$$

where $Mean(W_\tau)$ is the average total wealth and $Var(W_\tau)$ is the variance of the total wealth across the scenarios at the end of period τ. Parameter π indicates the relative importance of variance as compared to the expected value. This objective function leads to an efficient frontier of wealth at period τ by varying π between 0 and 1. To find a point on the efficient frontier, the investor finds a solution in which the marginal rate of substitution for expected value versus risk equals the negative slope of the efficient frontier.

An alternative to mean variance is the von Neumann–Morgenstern (VM) expected utility of wealth at period τ. Here, the objective becomes

$$\text{Maximize} \quad \sum_s p_s \, Utility(W_\tau^s), \tag{12}$$

where p_s is the probability of scenario s, W_τ^s is the wealth in period τ under scenario s, $Utility(.)$ is the VM utility function in [16]. The mean-variance model and the expected utility model are equivalent under certain conditions on the distribution of returns and the shape of the utility function [17,18]. Risk-averse decision-makers employ concave VM utility functions.

A third objective extends the expected utility model. Most investors are interested not only in their wealth at the end of the planning horizon, but also they prefer one set of trajectories over another—even when the results at the horizon are identical. Example is the rapid attainment of a certain level of wealth and the subsequent flattening of the wealth curve, as compared with a trajectory in which the growth in wealth comes at the end of the planning horizon. Given these two paths, most investors prefer the former. Thus, we ought to model intermediate preferences and the path to achieve a target wealth. In order to accomplish this goal, we employ a multi-objective formulation rather than the single expected utility function of wealth at the beginning of period τ. Several dimensions are added to the certainty equivalence questions, corresponding to the investor's wealth at several key junctures during the planning period. This extension is clearly a multi-objective problem, which can be addressed using approaches discussed in the literature. A general objective function for this problem is

$$\text{Maximize} \quad \sum_{k=1}^{K} \omega_k z_k, \tag{13}$$

where $\omega_k > 0$ and $\sum_{k=1}^{K} \omega_k = 1$ and z_k equals the objective function value for the kth attribute. Note that the objective functions will depend upon the entire wealth paths across the scenarios. The recommended course for implementing this path-dependent multi-criteria problem is an issue for future research.

The complete multi-stage financial planning model is to maximize Eq. (11) or Eq. (12) subject to the restrictions implied by Eqs. (1)–(10). In most instances, the model forms a nonlinear program possessing linear constraints. We emphasize the nonlinearities in the objective function. Most responsible decision-makers are risk averse—with concave utility functions. It seems unwise therefore to use a linear program to model these problems, especially when efficient nonlinear programming algorithms are now available for solving the resulting NLPs. More is said about the computational issues relating to nonlinearity in Sections 2 and 3.

1.2. Why a multi-period framework?

Once the computational issues are eliminated, the multi-stage financial planning system has several inherent advantages over single-period myopic models (see, e.g., Mulvey and Vladimirou [19]):

1. Consideration of *transaction costs* reflecting commission, fees, and other expenses incurred in trading activities. These costs are generally ignored in investment studies partially because of the

difficulty in measuring the degree of this factor and the consequential increase in computational complexity. In the real world, these costs can be substantial.

2. Ability to account for the decision-maker's *risk bearing attitudes*. The investor can define a preference structure that can be employed successfully over a number of years. One of the advantages of a systematic approach to investing is consistency—year after year maintaining a unified plan of action. It does the investor no good to become conservative after a dramatic drop in price—for example, selling all stocks! This behavior is easy to understand from a psychological standpoint, but makes poor sense for long-term investors. A multi-year financial planning system provides an opportunity for the investor to look both at the long- and short-term consequences of today's investment decisions.
3. An adequate treatment of *uncertainty* in important parameters, including any external cash flows and uncertain returns, so as to ensure that budget and liquidity requirements are met over time and that opportunity costs under various economic conditions are properly assessed.
4. Consideration of assets and liabilities in a *single integrated model* that addresses the whole financial planning problem and complies with accounting practices.
5. Ability to capture other factors involved in *practical decision-making*, including growth and budgetary requirements, as well as legal, institutional or policy provisions pertinent to the investor's problem.

1.3. Modeling stochastic parameters

The stochastic parameters that are needed for the financial planning model can be placed in three groups: (1) a small set of economic factors; (2) projected returns for the asset categories as implied by the values of the economic factors in the prior group; and (3) projected liabilities based on the implied values of the same economic factors. Again, the notion of a scenario is critical. A scenario consists of a complete and consistent set of parameters across the extended planning horizon T as required by the constraints in the financial planning model.

There are several goals to keep in mind when building a model for the stochastic parameters [20]. First, the procedures must be based on sound economic principles. For instance, the interest rates must be consistent with the returns for the fixed-income asset categories. International investments should be designed so that the foreign currencies are a separate category—for the purposes of developing reasonable hedging strategies. The basic trends should be preserved whenever possible—such as mean reversion in interest rates over an extended horizon. The projections should be evaluated with regard to their fit with historical data and trends.

A second goal is to design a stochastic modeling system flexible so that the system can be tailored to individual investor's circumstances. An investor will not trust the recommendations of a planning system unless the investor's general beliefs are properly portrayed in the stochastic models. In this regard, the model should be simple enough so that the investor can understand the basic philosophy and the key linkages among the modeling components. An understandable model will go a long way toward gaining the confidence of the investor—thus increasing the chances that a financial planning model will be employed.

The primary aim of scenario generation is to construct a number of scenarios that provide a reasonable representation of the universe of possible outcomes. This objective is much different than the generation of a single scenario, say for forecasting and trading strategies. Rather, we are interested in constructing a *representative* set of scenarios. In this regard, we must include scenarios that are both optimistic and pessimistic—of course, within a general modeling framework. Towers Perrin (one of the largest actuarial firms in the world) undertook such an effort. Their scenario generation process is called CAP:Link for capital market projections [5]. The process entails a cascading set of submodels, starting with the interest rate component. Towers Perrin uses a version of the two-factor interest rate model by Brennan and Schwartz [21]. The other submodels are driven by the interest rates and other economic factors. Dert [28] presents an excellent discussion of scenario development in the context of asset-liability management. Various components of the scenario generator are discussed, along with the time series models for generating price inflation, wage inflation, and asset returns.

Dupacova [22,23] presents post-optimality analysis via the contamination technique to provide a tool for investigating the changes in the selected scenarios and their probabilities on the optimal solution. Stochastic programming approach to asset allocation assumes a given discrete distribution for scenarios. The origin of scenarios, however, can be very diverse. They may follow a truly discrete distribution or are obtained from an approximation scheme. They may also be generated through some preliminary analysis of the problem in conjunction with subjective opinions of experts. Regardless of how they were generated, the robustness of the generation procedure is an important issue that cannot be ignored. The scenario generation should be robust in the sense that small perturbations of the chosen scenarios and of their probabilities should alter the outcome only slightly.

1.4. Integrating assets and liabilities

A key feature of the proposed multi-stage investment system is the attention given to integrating assets, liabilities, and investment goals. Most investors make investment decisions without reference to liabilities or investment goals (e.g., [24–26]). Focusing on assets alone misrepresents the risks and relative rewards to investor wealth for dynamic investment strategies, especially for investors who are risk averse. The key formula for wealth is

$$Wealth = \sum_i assets - PV(liabilities), \tag{14}$$

where the present value (PV) is required since many liabilities are not marketable. Likewise, in the case of investment goals, such as purchasing a retirement annuity, a fair value calculation must be performed in order to estimate the surplus wealth. Employing these ideas, Mulvey [27] extended the mean variance and the expected utility model to address liabilities in the context of asset allocation strategies.

There are several noteworthy implementations of asset-liability investment systems (e.g., [4,7,12, 28–31]). One of the largest is the Russell–Yasuda Kasai effort [4]. A multi-stage stochastic program assists the Yasuda Company in making investment decisions and analyzing overall risks. The Japanese insurance industry must abide by restrictions on the dividends and other rules. The objective function combines expected profit with piece-wise linearized penalties for violating specified targets with a specified probability. The stochastic programs are solved using IBM RS/6000 Model 530 workstations in conjunction with IBM's LP software package OSL. The largest problem consisted of 2048 scenarios. The initial delivered software system handles 256 scenarios. The Yasuda system fits the definition of a stochastic program since there are no explicit control rules for managing the rebalancing decisions at each stage during the planning horizon. Scenarios are generated via the standard scenario tree structure. The system was compared with an asset-only Markowitz model (the system that was in use beforehand) and was shown to be worth over $79 million in improved financial management. Other advantages are cited, such as enhanced communications. The system has been revised by the Russell company for a second version.

The second example involves the worldwide benefit consulting company Towers Perrin [5]. The objectives of their asset-liability investment system are to provide actuarial advice regarding the soundness of pension plans and to render recommendations as asset consultants. The Towers Perrin system depends upon dynamic stochastic control, as compared with the stochastic program employed in the Russell system. Rebalancing rules are restricted to the fixed-mix category. The Towers Perrin staff devoted considerable effort in the scenario generation process. The scenario generation program assists both actuaries for setting return assumptions as well as asset consultants who make recommendations concerning the risk and rewards for investment strategies. The dynamic stochastic control model is solved using PCs in conjunction with GRG nonlinear programming software [32]. The system handles 500–1000 scenarios. Due to the nature of the control framework, there is no need to generate scenarios in a tree structure. Rather, the scenarios consist of single sets of outcomes without any branching—equivalent to a tree with a single level. A global CAP:Link system is under development that incorporates interest rates, currency exchange rates, and stock market returns.

The third example involves a bank asset and liability model by Kusy and Ziemba [12]. They developed a multi-period stochastic linear programming model that includes a host of bank-related policy considerations and their uncertainties. A version of this model was tested for the Vancouver City Savings Credit Union for a 5-year planning period. The results indicated that the asset and liability system was theoretically and operationally superior to a deterministic counterpart. The computational requirements for the system were also comparable to those of the deterministic model.

2. Computational experiment

We show that the computational issues become critical to the success of the proposed financial planning system. First, the problems are enormous by current computer standards, even supercomputers; we must employ high performance computers, including parallel and distributed machines, in many situations. The difficulty resides in the nature of the dynamics of investing. Not only must decisions be made today, but also future conditional decisions must be considered. The number of conditional decisions quickly grows resulting in very large optimization problems. Second, most investors are risk averse when substantial amounts of money are involved, requiring nonlinear objectives or approximations therein. Large nonlinear programs have been difficult to solve.

This section reviews solution algorithms for the multi-stage financial planning model and its specializations. We focus on the solution of multi-stage stochastic programs possessing discrete-time decisions with a modest number of scenarios—typically under 1000–4000—and nonlinear objective functions for addressing risk aversion. For purposes of this section, we define the following nonlinear program to represent the financial planning model:

$$
\begin{aligned}
\text{[NLP]} \quad & \text{Maximize} \quad && f(x), \\
& \text{subject to} \quad && Ax = b, \\
& && x \geqslant 0,
\end{aligned}
$$

where f is a continuously differentiable objective function, A is an $[m \times n]$ matrix of coefficients, b is an $[m \times 1]$ vector of right-hand-side coefficients, and x is an $[n \times 1]$ vector of decision variables.

2.1. Direct solvers

We solve NLP via an enhancement to an interior point direct solver LOQO [8]. In the direct solution of NLP via interior-point methods, the primary computational step is the factorization of the normal equations ADA^t by means of the Cholesky (LL^t) method [33]. One of the major difficulties that arise when applying this approach to stochastic optimization is the amount of fill-in that occurs during factorization. Instead of factorizing a symmetric positive definite matrix ADA^t, Vanderbei and Carpenter [34] propose an alternative known as the KKT system. They factor the following matrix:

$$
\begin{bmatrix} -(Q + ZX^{-1}) & A^t \\ A & 0 \end{bmatrix},
$$

where Q is the Hessian matrix of f at the current iterate, Z is a diagonal matrix with the values of the dual slacks on the diagonals, and X is a diagonal matrix with the values of the primal values on its diagonal. Although this system is indefinite and larger than the normal equations, efficient methods exist for factoring it. In fact, strong arguments favor this matrix over the normal equations when solving nonlinear programs. Also, the reduced KKT system curtails some of the problems with dense columns and is employed by LOQO.

The tree structure of multi-stage stochastic program causes difficulty for the standard pre-ordering methods as employed by CPLEX [35] and LOQO [8]. In particular, the multiple min-degree procedure gives rise to considerable fill-in. Berger et al. [36] have developed an alternative ordering heuristic, called tree dissection, for the structure of multi-stage stochastic programs. We tested two variations of tree dissection in conjunction with LOQO. Preliminary test results are presented in Tables 1 and 2. These tests were conducted using a single Silicon Graphics workstation (R8000) at SGI headquarters in Mountain View, California on a six-stage financial investment problem. Table 1

Table 1
Floating point operations to factor KKT system (millions of operations)

Scenarios	LOQO	CPLEX	Full tree	N-2 levels
64	61	30	61	8
512	2348	11,327	351	70
1024	23,882	NA	781	179
2048	38,040	NA	1564	353
4096	152,873	NA	3131	721

Table 2
Computational tests with tree dissection (six-stage investment problem)

Senarios	Time (s)	LOQO iterations
512	448	31
1024	1477	49
2048	5138	73
4096	9991	70

illustrates the advantages of employing the tree dissection algorithm. Note that CPLEX was able to solve problems with 64 and 512 scenarios, but was unable to solve large problems due to memory and time limitations. The full-tree approach (column 4) improved upon computational performance, resulting in a linear relationship between the run-time (and operations) and the number of scenarios. A specialized version of tree dissection (called N-2 levels) was then developed, resulting in a further increase in efficiency. We are now able to solve stochastic problems with linear-time complexity.

Numerical analysts and others have developed efficient *parallel* Cholesky methods for factorizing sparse matrices. A critical idea is the supernode—a set of adjacent columns with the property of a dense triangular block on the diagonal and identical nonzero pattern for each column in the supernode below the diagonal. The structure of supernodes can be capitalized on. However, the supernode concept has not proven very useful when implementing the algorithms in a parallel environment since supernodes grow too large for an assignment to any single processor. Recently, Rothberg [37] and Rothberg and Gupta [38] developed an extremely efficient method for carrying out the computations. Rothberg divides the supernode into contiguous partitions, called panels, and then merges this idea with a block fan out approach. The primary idea is to map blocks to processors in a manner so that communications are minimized and such that the amounts of dense block calculations are done as independently as possible. The mapping is done by scatter decomposition.

Rather than describing the full algorithmic details, we simply show the results of preliminary tests with a large multi-stage investment problem using Rothberg's algorithm. The results are in general agreement with those of Jessup et al. [39] and Yang and Zenios [40]. The tests were conducted on an Intel paragon machine using a maximum of 128 processors. We generated the ADA^t matrix for a six-stage stochastic program of the financial planning problem for some larger examples. A single scenario consists of 156 variables and 96 constraints. The results of Rothberg's factorizations are listed in Table 3. Here, the factorization algorithm was specialized to address a parallel computer architecture. Note that the linear speed up occurred with a large number of scenarios for large-size problems.

These experiments are encouraging and especially so when coupled with the tree dissection concepts. They provide initial evidence that parallel direct solvers might be able to handle stochastic programs with over 10,000 scenarios within several minutes of run time in a parallel computer environment. If so, interior-point direct solvers should be employed whenever it is practical. The number of iterations stays quite low—generally under 50–100 steps—and the problem domain continues to be studied by many researchers. At times, the nonlinear programs are so large or the solution must

Table 3
Factorization run time (s) (N/A: not available)

Scenarios	Rows	Nonzero in A	Operations to factor
256	37,376	298,496	747M
512	74,752	596,992	5940M
1024	149,504	1,193,984	3073M

No. of processors	Scenarios		
	256	512	1024
16	3.11	14.53	N/A
32	1.80	8.61	6.24
64	1.23	4.59	3.82
128	N/A	N/A	2.25

be found in such a short period of time that direct solvers are infeasible. Decomposition algorithms are then the avenues to pursue.

2.2. Decomposition algorithms

Substantial progress has been made in the design of efficient decomposition algorithms for solving multi-stage stochastic programs since the original L-shape proposal by Van Slyke and Wets [41]. These algorithms take advantage of the stochastic program's structure (e.g., [42–49]). In several cases, implementation has occurred in a parallel or distributed computing environment. Motivating the use of decomposition algorithms, we note that the size of stochastic programs quickly grows as a function of the number of scenarios. As an example, our generated six-stage financial planning problem with 156 variables and 96 constraints per scenario grows to a problem with 100 scenarios—beyond the range of most current NLP solvers. Nevertheless, we can take advantage of the partially separable structure of the expected utility function

$$\text{Maximize} \quad f(x) = \sum_{s \in S} p_s U(x^s), \tag{15}$$

where $U(\cdot)$ is the VM function. This function readily splits across arbitrary subsets of scenarios. In the case of parallelization, we find that the expected utility model is easier to handle than the mean variance model which does not have a ready decomposition.

We present some preliminary computational results for the diagonal approximation algorithm (DQA) of Berger et al. [50]. The algorithm is based on the augmented Lagrangean function, where the nonanticipativity constraints are placed in the objective function. Suppose that the A matrix is split into two sets of constraints A_1 and A_2 where the latter represents the nonanticipativity constraints (or a subset therein). The augmented Lagrangean function is given by

$$L(x, \theta, \sigma) = f(x) + \theta(b_1 - A_2 x) + \sigma \|b_2 - A_2 x\|^2, \tag{16}$$

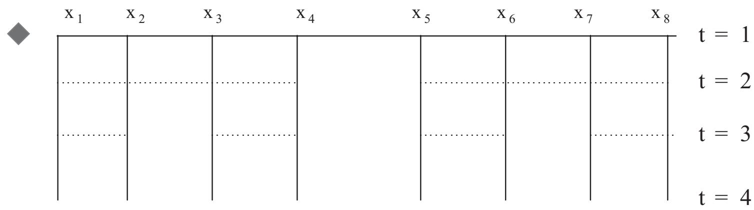

Fig. 3. A scenario tree and parallel data structure.

where θ and σ are the Lagrangean multipliers. The DQA algorithm approximates the Lagrangean at the current iterate by a quadratic and separable term. The main steps of the algorithm consists of dual variable update and the approximation of function $f(x)$. In terms of parallelization, the DQA algorithm splits the stochastic program into an arbitrary partition based upon the hardware configuration. The goal of the decomposition is to match the software algorithm to the available hardware. As mentioned, DQA forms an augmented Lagrangean function by dualizing nonanticipativity constraints. But not just any of the nonanticipativity constraints. Instead, most of the nonanticipativity constraints are kept explicit or avoided by using the compact formulation; the choice depends upon the capability of the solver with respect to dense columns. Subproblems are kept as large as possible in order to reduce total iteration count. The ideas can be illustrated with the example shown in Fig. 3. The number of scenarios equals eight with four time stages. Horizontal lines at stages 2–4 represent the nonanticipativity constraints. The scenario tree can be subdivided in an arbitrary fashion with the DQA algorithm—depending upon the hardware architecture available. For instance, we could break the problem into eight subproblems, one for each scenario x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8. Alternatively, we could divide the problem into two parts: x_1–x_4, and x_5–x_8. This strategy may be preferable if the subproblems can be solved in an efficient manner.

An important issue is matching the decomposition to the computer hardware. Suppose that two powerful processors are available and we wish to solve a problem with 1000 scenarios. We might form two subproblems of 500 scenarios—sending one to each processor. However, the processors may not be able to handle efficiently an NLP with 500 scenarios. Then we would further split the model into four subproblems of 250 scenarios. The objective is to carry out as little decomposition as possible while staying within the limits of efficient solution time via the solvers for the subproblems.

This flexible decomposition strategy can be highly effective. To give some idea of the efficiency of the convex-DQA algorithm, for example, we plot solution time as a function of the number of scenarios for the financial planning example (Fig. 4). Fig. 4 shows that multi-stage stochastic problems can be solved efficiently as the number of decision variables increases. This result occurs for both linear and convex objective functions. It is important to achieve linearity due to the large number of decision variables in stochastic programming problems. The largest problem in this domain consists of 3072 scenarios—corresponding to 480,000 total decision variables and almost 295,000 linear constraints (not including the nonanticipativity constraints). Although these problems remain difficult, integrative financial models are generally applied to long-term planning—hence, execution time has less concern so long as a number of test examples can be solved in a reasonable period. Investors must get an idea of the range of recommendations under differing sets of assumptions.

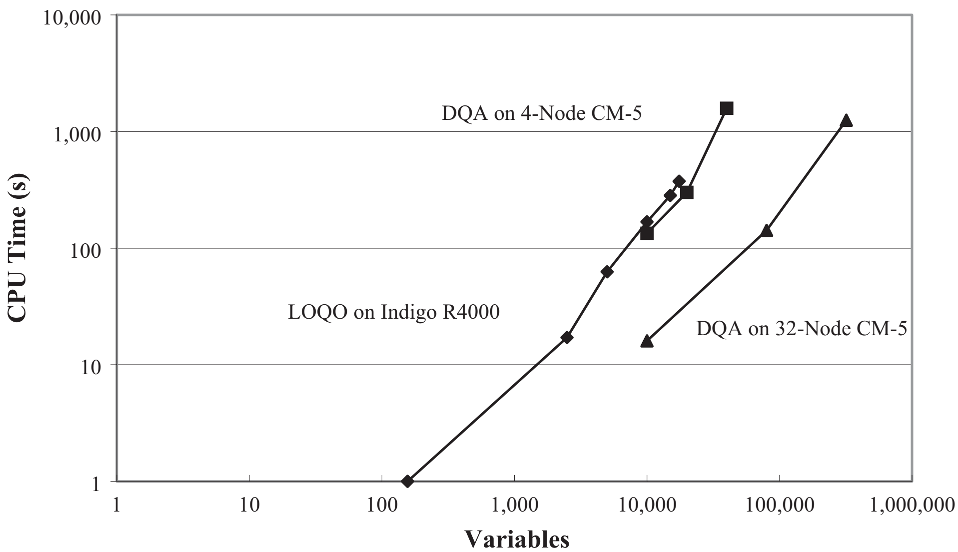

Fig. 4. Solution times for the convex DQA algorithm (six-stage financial planning system).

Investors interested in real-time planning can employ massively parallel computing. Likely, in the next 3–5 years, we will be able to solve substantial stochastic programs using desktop computers.

3. Dynamic stochastic control models

An alternative formulation for the multi-stage financial planning problem is to pose the model using a dynamic stochastic control framework. This framework has been used extensively in the finance literature (e.g., [51–53]). While the stochastic programming approach considers a small number of possible states or scenarios at a few points in time, the stochastic control approach considers a continuum of states described by a small number of state variables at each point in time. The control approaches attempt to find closed-form solutions—in continuous time or in discrete time—or they employ dynamic programming. The relative pros and cons of these competing methods have not yet been fully explored.

In Section 1 we briefly discussed the fixed-mix strategy, which is a special case of dynamic stochastic control. Constraint (9) renders the resulting model nonconvex due to nonlinear dynamics at each decision stage. While a search for the global optimum in the presence of nonconvexity is often difficult, efficient algorithms are available for the proposed class of multi-stage investment problems. Realistic size problems have been solved using these algorithms within a modest amount of computing on a workstation [54]. Below we provide a discussion of a global algorithm. For convenience, the maximization of f is replaced by the minimization of $g = -f$. In the context of the fixed-mix problem, the decision variables are λ_i for $i \in I$ and the constraints take the form

$$\sum_{i \in I} \lambda_i = 1. \tag{17}$$

The global optimization algorithm is based on a convex lower bounding of the original objective function g and the successive refinement of converging lower and upper bounds by means of standard branch and bound procedure. A convex lower bounding LB of g can be defined by augmenting g with a separable convex quadratic function of λ_i as proposed in [54]

$$LB = g + \alpha \sum_{i \in I} (\lambda_i^{\mathrm{L}} - \lambda_i)(\lambda_i^{\mathrm{U}} - \lambda_i), \tag{18}$$

where

$$\alpha \geqslant \max\{0, \max_{\substack{i \\ \lambda_i^{\mathrm{L}} \leqslant \lambda_i \leqslant \lambda_i^{\mathrm{U}}}} (-\tfrac{1}{2} eig_i^f)\}.$$

Note that $\lambda_i^{\mathrm{L}}, \lambda_i^{\mathrm{U}}$ are the lower and upper bounds of λ_i initially set to $\lambda_i^{\mathrm{L}} = 0$ and $\lambda_i^{\mathrm{U}} = 1$ Also, α is a nonnegative parameter which must be greater or equal to the negative one-half of the minimum eigenvalue of g over $\lambda_i^{\mathrm{L}} \leqslant \lambda_i \leqslant \lambda_i^{\mathrm{U}}$. The parameter α can be estimated either through the solution of an optimization problem or by using the concept of the measure of a matrix. The effect of adding the extra term to g is to make LB convex by overpowering the nonconvexity characteristic of g with the addition of the term 2α to all of its eigenvalues:

$$eig_i^{LB} \quad = eig_i^g + 2\alpha \quad \text{for } i \in I.$$

Here eig_i^{LB} and eig_i^g are the eigenvalues of LB and g, respectively. This function LB defined over the box constraints $[\lambda_i^{\mathrm{L}}, \lambda_i^{\mathrm{U}}]$ for $i \in I$ involve a number of properties that enable us to construct a global optimization algorithm for finding the global minimum of g.

The tactical details of the global algorithm, such as methods for partitioning the feasible region, are described in [54,55]. This procedure has proven to be effective in several problem domains and has been recently applied successfully to a practical fixed-mix investment problem with over 500 scenarios, 8 asset categories, and 20 time periods. Good solutions were obtained within several minutes or less on HP workstations. The basic approach readily extends to a variety of control strategies.

There has not been much research comparing stochastic programming and the stochastic control framework. Several issues arise when considering either approach. A control strategy is easy to understand and implement and can be readily tested with out-of-sample scenarios. The number decision variables resulting from a fixed-mix strategy are restricted to some small multiple of the number of asset categories and therefore exempt from the dimensionality curse. Unfortunately, the reduction in problem size introduces nonconvexity, which makes the search for a global recommendation extremely complicated. A stochastic programming, on the other hand, may provide a much better recommendation as it is able to handle a variety of constraints as well as nonanticipativity. The difficulty, however, is in that many investors are unable to execute large stochastic programs at each period possibly involving thousands of scenarios. Further testing is needed in order to understand the relative pros and cons of these two approaches.

4. Conclusions and future directions

The proposed multi-stage financial planning model provides a general framework for integrating all asset and liability decisions for a large financial entity—such as an insurance company, bank, or pension plan—as well as for individual investors. This comprehensive approach measures the risk and rewards of alternative investment strategies. Without an integrative asset-liability model, investors are unable to properly measure *risks* to their wealth. The usual asset-only approach inadequately evaluates the impact of investments on wealth and achieving investment goals. The main lesson is that investment models must be tailored to individual circumstances. The multi-stage stochastic program provides an ideal vehicle for developing a financial plan that fits the investor's needs.

Although the resulting models are large convex stochastic programs or nonconvex dynamic control problems, efficient solution algorithms are now becoming available. The sharp decline in computer costs and the commensurate increase in usability improves prospects that financial optimization will become commonplace in large financial organizations. It should become a critical tool for individual investors as well. To this end, nonlinear objectives for modeling risk aversion are a standard element.

Future research should continue along several dimensions. First, we must increase the size of solvable stochastic programs so that additional scenarios can be handled in a practical fashion. There is no fundamental reason why we cannot address 10,000 to 100,000 scenarios using parallel and distributed computers. Certainly, the raw computing power will be available. Whether or not direct solvers or decomposition algorithms are best is a matter for future research.

Another computational issue involves generating scenarios. In particular, out-of-sample testing will be critical in order to compute valid bounds on the model recommendations. When it applies, dynamic stochastic control can be useful. The control system assists in the selection of the scenarios—for instance, by generating importance estimates for adding (or deleting) scenarios as they affect the solution to the control problem. These scenarios should be linked to the stochastic program. Of course, embedding a stochastic program within a simulation system such as carried out by Worzel et al. [6] to evaluate the precision of the recommendations is possible. The approach requires large computational resources and may be impractical. Linking simulation and optimization models, however, will be increasingly important, as multi-stage stochastic programs become more widespread in practice.

There are a number of algorithmic items to explore. One is to take further advantage of the structure of the multi-stage stochastic program within a parallel interior-point algorithm. For instance, we can conduct the Cholesky factorization using modern sparse matrix calculations on parallel or distributed computers. Rothberg's approach [37] seems to be a potential winner. Our preliminary experiments in Section 3 show that large matrices can be quickly factored. Jessup et al. [39] have conducted similar tests on the CM-5 computer. Solving the stochastic program as quickly as possible will increase the chances that individual investors and institutions will apply the models. In the case of decomposition methods, the sparse matrix calculations are key for techniques such as DQA which use an interior-point algorithm for solving subproblems. Any substantial progress on this issue leads to immediate gains in the decomposition algorithm. Also, the restarting issue for interior-point algorithms remains.

As an upcoming topic, we must soon consider the near simultaneous solution of hundreds or thousands of multi-stage stochastic programs. If individual investors begin to employ long-term financial plans based on stochastic optimization, there will be a need for hardware and software that can handle this unprecedented computational requirement. Parallel approaches could be applied, for example, by constructing the models such that the nonzero matrix structure is identical across individuals with different coefficient values. This problem will tax the largest supercomputers. But if successful, the multi-stage financial planning systems for individuals have enormous potential for improving the way that individuals manage their financial affairs.

Acknowledgements

This research was supported in part by NSF Grant (CCR-9102660), Towers Perrin, and Proprietary Financial Products, Inc. We are grateful for Adam Berger's assistance with the computational tests.

References

[1] Zenios SA, editor. Financial optimization. Cambridge: Cambridge University Press, 1993.
[2] Zenios SA, Ziemba WT, editors. Theme issue on financial modeling. Management science 1992;38:1525–1528.
[3] Mulvey JM, Rosenbaum DP, Shetty B. Strategic financial risk management and operations research. European Journal of Operational Research 1997;97:1–16.
[4] Cariño DR, Kent T, Myers DH, Stacy C, Sylvanus M, Turner A, Watanabe K, Ziemba WT. The Russell–Yasuda Kasai model: an asset liability model for a Japanese insurance company using multi-stage stochastic programming. Interfaces 1994;24:29–49.
[5] Mulvey JM. Generating scenarios for the towers perrin investment systems. Interfaces 1995;26:1–15.
[6] Worzel K, Vassiadou-Zeniou C, Zenios SA. Integrated simulation and optimization models for tracking fixed-income indices. Operations Research 1995;42:223–33.
[7] Golub B, Holmer M, McKendall R, Pohlman L, Zenios SA. A stochastic programming model for money management. European Journal of Operational Research 1995;85:282–96.
[8] Vanderbei RJ. LOQO User's manual. Technical Report SOR-92-5, Department of Civil Engineering and Operations Research, Princeton University, Princeton, NJ, 1992.
[9] Bradley SP, Crane DB. A dynamic model for bond portfolio management. Management Science 1972;19:139–51.
[10] Kall P, Wallace SW. Stochastic programming. New York, NY: John Wiley, 1994.
[11] Kallberg JG, White WR, Ziemba WT. Short term financial planning under uncertainty. Management Science 1982;28:670–82.
[12] Kusy MI, Ziemba WT. A bank asset and liability management model. Operations Research 1986;34:356–76.
[13] Mulvey JM, Vladimirou H. Solving multistage stochastic networks: an application of scenario aggregation. Networks 1991;21:619–43.
[14] Perold AF, Sharpe, WF. Dynamic strategies for asset allocation. Financial Analysts Journal 1988;44:16–27.
[15] Erza D. Asset allocation by surplus optimization. Financial Analyst Journal 1991;47:51–7.
[16] Keeney R, Raiffa H. Decisions with multiple objectives: preferences and value tradeoffs. New York: Wiley, 1993 (Reprinted by Cambridge University Press, Cambridge, MA, 1976).
[17] Markowitz HM. Portfolio selection: efficient diversification of investments, 2nd ed. Cambridge, MA: Basil Blackwell, 1991.
[18] Kroll Y, Levy H, Markowitz H. Mean variance versus direct utility maximization. Journal of Finance 1984;39: 47–62.
[19] Mulvey JM, Vladimirou H. Stochastic network programming for financial planning problems. Management Science 1992;38:1642–64.
[20] Mulvey JM, Rosenbaum DP, Shetty B. Parameter estimation in stochastic scenario generation systems. European Journal of Operational Research 1999;118:563–77.
[21] Brennan MJ, Schwartz ES. An equilibrium model of bond pricing and a test of market efficiency. Journal of Financial and Quantitative Analysis 1982;17:75–100.
[22] Dupacova J. Stochastic programming with incomplete information: a survey of results on postoptimization and sensitivity analysis. Optimization 1987;18:507–32.
[23] Dupacova J. Scenario based stochastic programs: resistance with respect to sample. Annals of Operations Research 1996;64:21–38.

[24] Perold AF. Large-scale portfolio optimization. Management Science 1984;30:1143–60.
[25] Solnik B, Noetzlin B. Optimal international asset allocation. Journal of Portfolio Management 1982;9:11–21.
[26] Sorensen E, Mezrich J, Thadani D. Currency hedging through portfolio optimization. Journal of Portfolio Management 1993;19:78–85.
[27] Mulvey JM. A surplus optimization perspective. Investment Management Review 1989;3:31–9.
[28] Dert C. A multi-stage chance constrained programming approach to asset-liability management for pension funds. Ph.D. Thesis, Erasmus University, Rotterdam, The Netherlands, 1995.
[29] Dempster MAH. The development of the midas debt management system. In: Ziemba WT, Mulvey JM, editors. World wide asset and liability modeling. Cambridge, MA: Cambridge University Press, 1996.
[30] Holmer MR. The asset/liability management strategy at Fannie Mae. Interfaces 1994;24:3–21.
[31] Holmer MR, Zenios SA. The productivity of financial intermediation and the technology of financial product management. Operations Research 1995;43:970–82.
[32] Smith S, Lasdon L. Solving large sparse nonlinear programs using GRG. ORSA Journal on Computing 1992;4: 2–15.
[33] Lustig J, Mulvey JM, Carpenter TJ. Formulating stochastic programs for interior point methods. Operations Research 1991;39:757–70.
[34] Vanderbei RJ, Carpenter TJ. Symmetric indefinite systems for interior-point methods. Mathematical Programming 1993;58:1–32.
[35] CPLEX, Using the CPLEX callable library and CPLEX mixed integer library. Technical Report, CPLEX Optimization Inc., Las Vegas, NV, 1993.
[36] Berger AJ, Mulvey JM, Rothberg E, Vanderbei R, Solving multi-stage stochastic programs using tree dissection. Technical Report SOR-96-05, Department of Civil Engineering and Operations Research, Princeton University, Princeton, NJ, 1992.
[37] Rothberg E. Performance of panel and block approaches to sparse Cholesky factorization on the iPSC/860 and Paragon multicomputers. Technical Report, Intel Supercomputer Systems Division, Beaverton, OR, 1993.
[38] Rothberg E, Gupta A. An efficient block-oriented approach to parallel sparse Cholesky factorization. Supercomputing 1993;503–12.
[39] Jessup ER, Yang D, Zenios SA. Parallel factorization of structured matrices arising in stochastic programming. SIAM Journal on Optimization 1994;4:833–46.
[40] Yang D, Zenios SA. A scalable parallel interior point algorithm for stochastic linear programming and robust optimization. Computational Optimization and Applications 1997;7:143–58.
[41] Van Slyke R, Wets RJ-B. L-shaped linear programs with applications to optimal control and stochastic programming. SIAM Journal of Applied Mathematics 1969;17:638–63.
[42] Birge JR, Donohue CJ, Holmes DF, Svintsitski OG. A parallel implementation of the nested decomposition algorithm for multistage stochastic linear programs. Mathematical Programming 1994;75:327–52.
[43] Birge JR, Holmes DF. Efficient solution of two-stage stochastic linear programs using interior point methods. Computational Optimization and Applications 1992;1:245–76.
[44] Dantzig GB, Infanger G. Multi-stage stochastic linear programs for portfolio optimization. Annals of Operations Research 1993;45:59–76.
[45] Gassmann HI. A computer code for the multi-stage stochastic linear programming problem. Mathematical Programming 1990;47:407–23.
[46] Higle JL, Sen S. Stochastic decomposition: an algorithm for two-stage linear programs with recourse. Mathematics of Operations Research 1991;16:650–69.
[47] Infanger G. Planning under uncertainty: solving large-scale stochastic linear programs. Danvers, MA: Boyd and Fraser, 1994.
[48] Nielsen S, Zenios SA. A massively parallel algorithm for nonlinear stochastic network problems. Operations Research 1993;41:319–37.
[49] Rockafeller RT, Wets RJ-B. Scenarios and policy aggregation in optimization under uncertainty. Mathematics of Operations Research 1991;16:119–47.
[50] Berger AJ, Mulvey JM, Ruszczynski A. An extension of the DQA algorithm to convex stochastic programs. SIAM Journal on Optimization 1994;4:735–53.
[51] Brennan MJ, Schwartz ES, Lagnado R. Strategic asset allocation. Journal of Economic Dynamics and Control, 1996;21:1377–1403.

[52] Chow GC. Optimal control without solving the Bellman equation. Journal of Economic Dynamics and Control 1993;17:621–30.
[53] Dixit AK, Pindyck RS. Investment under uncertainty. Princeton, NJ: Princeton University Press, 1994.
[54] Maranas CD, Androulakis IP, Floudas CA, Berger AJ, Mulvey JM. Solving stochastic control problems in finance via global optimization. Journal of Dynamics and Control 1997;21:1405–25.
[55] Floudas CA, Visweswaran V. Primal–relaxed dual global optimization approach. Journal of Optimization Theory and its Application, 1993;78:187–225.

John M. Mulvey is a Professor at the Department of Operations Research and Financial Engineering, Princeton University, Princeton, NJ. He is an expert in large-scale optimization models and algorithms, especially financial applications. He has implemented integrated risk management for many large companies, including American Express, Towers Perrin—Tillinghast, Pacific Mutual, and St. Paul Insurance.

Bala Shetty is a Professor at the Department of Information and Operations Management, Texas A&M University, College Station, TX. His primary research interests are in large-scale optimization and finance.