STOCHASTIC PROGRAMMING: OPTIMIZING THE UNCERTAIN

John R. Birge
Department of Industrial and Operations Engineering
The University of Michigan
Ann Arbor, MI 48109

Technical Report 92-24

March 1992

Stochastic Programming: Optimizing the Uncertain

John R. Birge*
Department of Industrial and Operations Engineering
The University of Michigan
Ann Arbor, Michigan, USA 48109

Abstract: Finding optimal decisions often involves the consideration of certain random or unknown parameters. When these uncertainties are built directly into an optimization model, the result is a stochastic program. We describe an example from financial planning that illustrates the usefulness of a stochastic program in terms of the value of the stochastic solution and the difference between this quantity and the expected value of perfect information. We also briefly outline methods for constructing stochastic programs that bound optimal values. We also discuss basic computational techniques and the advantages from considering the special structure inherent in stochastic programs.

Keywords: stochastic programming, approximations, finance, large-scale optimization.

* This author's work was supported by the National Science Foundation under Grant ECS 88-15101.

1. Introduction

Practical decisions often involve the consideration of uncertain or stochastic parameters. Optimization procedures are increasingly helpful as the size and complexity of tractable problems grow. Ignoring fundamentally random characteristics may, however, limit the usefulness of an optimal solution. Stochastic programs that explicitly consider randomness may be much more beneficial for actual operations.

The expected objective advantage of using a stochastic programming solution over a deterministic program solution is called the *value of the stochastic solution.* In Section 2, we illustrate this quantity through a model in financial planning in which one seeks to maximize expected utility concerning a child's higher education. Of particular note is that this quantity is different from the expected value of perfect information which represents the expected objective improvement possible with perfect knowledge of the future.

The major difficulty in stochastic programming is evaluating the future effect of a current action. Approximations are presented in Section 3 that can bound this future effect from above or from below. With these approximations, stochastic programs become large-scale mathematical programs that have a particular structure. We briefly describe some techniques to take advantage of that structure in Section 4.

2. Financial Planning and the Value of the Stochastic Solution

Financial decision making problems can often be modeled as stochastic programs. This area represents one of the largest application areas of stochastic programming. Many references can be found in, for example, Mulvey and Vladimirou [27] and Ziemba and Vickson [36].

We consider a simple example that illustrates additional stochastic programming properties. The random variables reflect uncertain investment yields. The role of the stochastic program is to hedge against poor outcomes by maximizing an expected objective function that is concave and represents some aversion to risk.

For the current problem, suppose we wish to provide for a child's college education Y years from now. We currently have $\$w$ to invest in any of K investments. After Y years, we will have a wealth of $\$W$ that we would like to have exceed the tuition goal of $\$G$. We suppose that we can change investments every y years so we have $T = Y/y$ investment periods. For our purposes here, we will ignore transaction costs and taxes on income although these considerations would be important in practice.

In formulating the problem, we must first describe our objective in mathematical terms. We suppose that exceeding $\$G$ after Y years would be equivalent to our having an income of i of the excess while not

meeting the goal would lead to borrowing for a cost q of the amount short. This gives us the concave utility function. Many other forms of nonlinear utility functions are of course possible. See Kallberg and Ziemba [23] for a description of their relevance in financial planning.

The major uncertainty in this model is the return on each investment k within each period t. We describe this random variable as $\mathbf{r}(k,t) = r(k,t,\omega)$ where ω is some underlying random element. The decisions on investments will be random as well. We describe these decisions as $\mathbf{x}(k,t) = x(k,t,\omega)$. From the randomness of the returns and investment decisions, our final wealth will also be a random variable $\mathbf{W} = W(\omega)$.

A key point about this investment model is that we cannot completely observe the random element ω when we make all of our decisions $x(k,t,\omega)$. We can only observe the returns that have already taken place. In stochastic programming, we say that we cannot *anticipate* every possible outcome so that our decisions are *nonanticipative* of future outcomes. Before the first period, this restriction corresponds to saying that we must make fixed investments, $x(k,1)$, for all $\omega \in \Omega$, the space of all random elements.

In the next period, we suppose that the random elements ω correspond to N_1 different possibilities for outcomes in the first investment period. We can therefore partition Ω into $\Omega_1^1, \ldots, \Omega_{N_1}^1$ corresponding to these different initial outcomes. Decisions for a given value of ω must be the same for every ω within the set. We can therefore describe our second period decisions just in terms of the Ω_i^1 that occurs in the first period. We write these decision variables as $x(k,2,i)$, where $i = 1, \ldots, N_1$.

We can continue this process by defining $\Omega_{i_1,\ldots,i_t}^t$ as the set of all ω that correspond to outcomes i_j in periods $j = 1, \ldots, t$. We can then describe the nonanticipative decisions as $x(k,t+1,i_1,\ldots,i_t)$, which depend on the outcomes $r(t,i_1,\ldots,i_t)$. In the following, for simplicity, we assume that each set of outcomes $i_1,\ldots,i_{t-1}$ up to time $t-1$ leads to N_t outcomes at time t. In all T periods, we would then have $N_1 \cdot N_2 \cdots N_T$ different possible outcomes. To illustrate how quickly these problems can grow, note that, with just ten outcomes per period, we would obtain 10^{10} outcomes in 10 periods.

We need to attach probabilities to all outcomes. We let the probability of the ith outcome in period t given outcomes i_j in periods $j = 1, \ldots, t-1$ be $p(t,i_1,\ldots,i_t)$. Note that we must have $\sum_{i=1}^{N_t} p(t,i_1, \ldots, i_{t-1}, i) = 1$ for all t and $i_1, \ldots, i_{t-1}$.

We then can state a formulation for our problem. We wish to find

$$
\begin{array}{rrll}
\max z = & \sum_{i_1,\ldots,i_T} p(T,i_1,\ldots,i_T)(iv(i_1,\ldots,i_T) & -qs(i_1,\ldots,i_T)) & \\
\text{s. t.} & \sum_{k=1}^{K} x(1,k) & = w, & \\
& \sum_{k=1}^{K} r(k,t,i_1,\ldots,i_{t-1})x(k,t-1,i_1,\ldots,i_{t-2}) & & \\
& -\sum_{k=1}^{K} x(k,t,i_1,\ldots,i_{t-1}) & = 0, & \\
& & \text{for all } (i_1,\ldots,i_{t-1}), & \\
& & t=2,\ldots,T; & \\
& \sum_{k=1}^{K} r(k,T,i_1,\ldots,i_T)x(k,T,i_1,\ldots,i_{T-1}) & & \\
& -v(i_1,\ldots,i_T)+s(i_1,\ldots,i_T) & = G, & \\
& x(k,t,i_1,\ldots,i_{t-1}) & \geq 0, & \\
& v(i_1,\ldots,i_T) & \geq 0, & \\
& s(i_1,\ldots,i_T) & \geq 0, & \\
& & \text{for all } 1 \leq i_t \leq N_t; & \\
& 1 \leq k \leq K; & 1 \leq t \leq T. &
\end{array}
\tag{2.1}
$$

where $v(i_1,\ldots,i_T)$ represents any amount above the goal and $s(i_1,\ldots,i_T)$ represents any shortfall from the target value G.

Another approach to multistage problems like this is to consider the possible outcomes over the entire horizon as *scenarios*, σ. We then substitute a scenario set S for the random elements Ω. Probabilities, $p(\sigma)$, returns, $r(k,t,\sigma)$, and investments $x(k,t,\sigma)$ become functions of the T-period scenarios.

We must still maintain nonanticipativity, but this time we do so explicitly in the formulation via constraints. First, the scenarios that correspond to the same set of past outcomes at each period form groups, $S^t_{i_1,\ldots,i_{t-1}}$, for scenarios at time t. Now, all actions up to time t must be the same within a group. We do this through an explicit constraint. The new formulation of (2.1) becomes:

$$
\begin{array}{rcll}
\max z = & \sum_{\sigma} p(\sigma)(iv(\sigma)-qs(\sigma)) & & \\
\text{s. t.} \quad \sum_{k=1}^{K} x(k,1,\sigma) & & = w, \forall \sigma \in S; & \\
\sum_{k=1}^{K} r(k,t,\sigma)x(k,t-1,\sigma) & -\sum_{k=1}^{K} x(k,t,\sigma) & = 0, \forall \sigma \in S; & \\
& & t=2,\ldots,T; & \\
\sum_{k=1}^{K} r(k,T,\sigma)x(k,T,\sigma) & -v(\sigma)+s(\sigma) & = G; & \\
(\sum_{\sigma' \in S^t_{I(\sigma,t)}} p(\sigma')x(k,t,\sigma')) & -(\sum_{\sigma' \in S^t_{I(\sigma,t)}} p(\sigma'))x(k,t,\sigma) & = 0, \forall 1 \leq k \leq K; & \\
& \forall 1 \leq t \leq T; \forall \sigma \in S; & & \\
x(k,t,\sigma) \geq 0, & v(\sigma) \geq 0, & s(\sigma) \geq 0; & \\
\forall 1 \leq k \leq K; & \forall 1 \leq t \leq T; & \forall \sigma \in S; &
\end{array}
\tag{2.2}
$$

where $I(\sigma,t) = \{i_1,\ldots,i_{t-1}\}$ such that $\sigma \in S^t_{i_1,\ldots,i_{t-1}}$. Note that the last equality constraint indeed forces all decisions within the same group at time t to be the same. Formulation 2.2 has a special advantage for the problem here because these *nonanticipativity* constraints are the only constraints linking the separate scenarios. Without them, the problem would decompose into a separate problem for each σ, maintaining the structure of that problem.

In modeling terms, this simple additional constraint makes it relatively easy to move from a deterministic model to a stochastic model of the same problem. The addition of the scenario indicators and nonanticipativity constraints are the only additions to a deterministic model. Given the ease of this modeling effort, standard optimization procedures can be simply applied to this problem. However, as we noted above, the number of scenarios can become extremely large. Standard methods may not be able to solve the problem in any reasonable fashion, necessitating other techniques.

In this financial problem, it is particularly worthwhile to try to exploit the underlying structure of the problem without the nonanticipativity constraints. This relaxed problem is in fact a generalized network that allows the use of efficient network techniques.

With either formulation (2.1) or (2.2), in completing the model, some decisions must be made about the possible set of outcomes or scenarios and the coarseness of the period structure, i.e., the number of periods T allowed for investments. We must also find probabilities to attach to outcomes within each of these periods. These probabilities are often approximations that can, as we shall see in Section 3, provide bounds on true values or on uncertain outcomes with incompletely known distributions. A key observation we will make is that the important step is to include stochastic elements at least approximately and that deterministic solutions will most often give misleading results.

To illustrate the effects of including stochastic outcomes as well as modeling effects from choosing the time horizon Y and the coarseness of the period approximations T, we use a simple example with $K = 2$ possible investment types, stocks (1) and government securities (bonds) (2). We begin by setting Y at 15 years and allow investment changes every 5 years so that $T = 3$.

The other data for this example include $N_1 = N_2 = N_3 = 2$, so we have $|S| = 8$ scenarios. The scenarios corresponding to independent and equal likelihoods of having (inflation adjusted) returns of 1.25 for stocks and 1.14 for bonds or 1.06 for stocks and 1.12 for bonds over the 5 year period. This yields probabilities $p(\sigma) = 0.125$ for each scenario. The returns are $r(1,t,\sigma) = 1.25, r(2,t,\sigma) = 1.14$ for $t = 1, \sigma = 1,\ldots,4$, for $t = 2, \sigma = 1,2,5,6$, and for $t = 3, \sigma = 1,3,5,7$. In the other cases, $r(1,t,\sigma) = 1.06, r(2,t,\sigma) = 1.12$.

The remaining data are the initial wealth, $w = 55,000$, the target value, $G = 80,000$, the surplus reward, $i = 5$, and the shortage penalty,$q = 20$. Solving the problem in (2.2) with these parameter values yields an optimal expected utility value of -7.6. We will call this RP, for the expected *recourse problem* solution. The optimal solution (in thousands of dollars) appears in Table 1.

In this solution, the initial investments is heavily in stocks ($41,500) with only $13,500 in the government securities. Notice the reaction to first period outcomes however. In the case of scenarios 1-4, stocks are even

Period, Scenario	Stocks	Govt Secs 1
1,1-8	41.5	13.5
2,1-4	65.1	2.17
2,5-8	36.7	22.4
3,1-2	83.8	0.0
3,3-4	0.0	71.4
3,5-6	0.0	71.4
3,7-8	64.0	0.0
Scenario	Above G	Below G
1	24.8	0.0
2	8.87	0.0
3	1.42	0.0
4	0.0	0.0
5	1.42	0.0
6	0.0	0.0
7	0.0	0.0
8	0.0	12.2

Table 1. Optimal solution with 3-period stochastic program.

more prominent while scenarios 5-8 reflect a more conservative government security portfolio. In the last period, notice how the investments are either completely in stocks or completely in bonds. This is a general trait of one-period decisions. It occurs here because in scenarios 1,2, there is no risk of missing the target. In scenarios, 3-6, stock investments may cause one to miss the target so they are avoided. In scenarios 7,8, the only hope of reaching the target is through stocks.

We now compare the results in Table 1 with a deterministic model. If we substitue expected returns for the random variables, then stocks, with an expected 5-year yield of 1.155, dominate bonds. Everything is invested in stocks in each period. If we apply this decision process to the stochastic model with the alternative outcomes and probabilities given above, then the expected utility value is -19.2, which we call EMV for the *expectation of the mean value solution*. The *value of the stochastic solution* (VSS, introduced in Birge [1]) is the difference between the expected utility of the stochastic programming model ($RP = -7.6$) and the expected value for using the deterministic model with means substituted for the random variables ($EMV = -19.2$). In this case, that difference is

$$VSS = RP - EMV = -7.6 - (-19.2) = 11.6.$$

The expected value of perfect information on the other hand compares the recourse problem value (or maximum expected utility, RP) to the expectation of solution values that would be obtained if the future was

known perfectly. In this problem, there are eight potential future outcomes (two per period). The perfect information (or, in stochastic programming, following Madansky [26] the *wait-and-see* solution) is to invest in whatever has the highest yield in each period. In this case, we would invest in stocks if stocks increase 25% in a five year period (bonds increase 14%) and we would invest in bonds if bonds increase 12% (stocks increase 6%). Thus, with probability 0.125, we receive a return of $(1.25)^3 = 1.953$, with probability 0.125, we receive a return of $(1.12)^3 = 1.405$, with probability 0.375, we receive a return of $1.12 * (1.25)^2 = 1.75$, and, with probability 0.375, we receive a return of $(1.12)^2 * 1.25 = 1.568$. The result is that we would have an expected utility of $WS = 52.5$. The expected value of perfect information is

$$EVPI = WS - RP = 52.5 - (-7.6) = 60.1.$$

In this case, $EVPI > VSS$, but in many cases (see Birge [1]), we may have $VSS > EVPI$. In fact, since $WS > RP > EMV$, VSS and $EVPI$ are only assured of being the same when $WS = EMV$. Either could be zero, while the other is positive.

As we stated earlier, the model could also be divided into a coarser (or finer) period structure or we could use a shorter time horizon. In either of these cases, the results may be substantially different. For example, suppose we only allow two periods of 7.5 years each. We adjust the probabilities (to maintain mean yields and to reflect some decreased variance) so that the 7.5 year yields are either 1.37 or 1.13 for stocks and 1.22 or 1.18 for bonds, again with equal probabilities for the higher or lower pair in each period. In this case, we obtain the solution in Table 2. Notice now that all first period investment is in stock as in the mean value solution. If we implement this solution, then in 5 years, we would have \$68,750 with probability one half and \$58,300 with probability one half. We could then solve another two period problem to determine optimal investments from that point on. The result is an expected utility of 55.704 in the first case, and -75.335 in the second case. The overall expected utility of using this two-period rolling horizon type of solution is -9.816. The value of using a three-period model over this procedure is then $9.816 - 7.57$ or 2.25. This type of model sensitivity can be useful in determining how finely to divide a time interval when only one decision is implemented before another model is solved.

Another option for formulating a simpler model is to limit the horizon. This may seem quite reasonable since indeed we are only interested now in the first period decision. Consider then a two-period, 10-year model. In this case, we need to determine conditions on the end of the horizon to make our shorter time horizon model reflect the original model as closely as possible. This is the problem of mitigating *end effects*. It seems reasonable here to choose a target value that will ensure our achieving the original \$80,000 target value at year 15. We, therefore, make our 10 year target equal to \$71,400. Solving this two-period, 10 year problem with all other data as in the original 3 period model, we obtain the optimal solution in Table 3.

Period, Scenario	Stocks	Govt Secs
1,1-4	55.0	0.0
2,1-2	75.4	0.0
2,3-4	27.9	34.3
Scenario	Above G	Below G
1	23.23	0.0
2	5.15	0.0
3	0.0	0.0
4	0.0	8.05

Table 2. Optimal solution with 2-period, 15-year stochastic program.

Period, Scenario	Stocks	Govt Secs
1,1-4	9.8	45.2
2,1-2	0.0	63.8
2,3-4	17.1	44.0
Scenario	Above G	Below G
1	1.28	0.0
2	0.0	0.0
3	0.0	0.0
4	0.0	4.12

Table 3. Optimal solution with 2-period, 10-year stochastic program.

Notice how different the 10 year solution is from the other solutions. We now predominantly invest in bonds in the first period because we have not fully considered the chances of recovering later from initial poor stocks investments. In this case in five years, with probability one half we have \$63,800, and, with probability one half, we have \$61,100. Again, we may solve the models with two five-year periods at these points. The result is an expected utility of 9.342 in the first case and -28.845 in the second case. The overall expected utility is -9.752. This expected utility is quite close to that obtained with the longer period two-period model, but the variance of the utility is much less.

All three formulations demonstrate how relatively minor changes in formulations can lead to great differences even in first-period decisions. The goal is of course to create models that allow for some robustness in response to minor changes.

In closing this section, note that the mathematical form of this problem actually represents a broad class of control problems. In fact, it is basically equivalent to any control problem governed by a linear system of differential equations. We have merely taken a discrete time approach to this problem. This approach can be applied to the control of a wide variety of electrical, mechanical, chemical, and economic systems. We merely redefine state variables (now,wealth) in each time period and the controls (investment levels). The random gain or loss is reflected in the return coefficients. Typically, these types of control problems would have nonlinear (e.g., quadratic) costs associated with the control in each time period. This presents no complication for our purposes, so we may include any of these problems as potential applications.

In this problem, we had a limited set of possible outcomes in each period. In practical problems, we could not expect such a small finite set of realizations. One of the main steps in stochastic program modeling is how to select such sets of scenarios and how to compare their use with what may actually happen. The goal is then to create approximations with a small number of realizations that bound the expected objective value from above and below. We discuss the main form of these bounds in the next section.

3. Discrete Bounding Approximations

The most common procedures in stochastic programming approximations are to find some relatively low cardinality discrete set of realizations that somehow represents a good approximation of the true underlying distribution or whatever is known about this distribution. The basic procedures are extensions of Jensen's inequality ([21], generalization of the midpoint approximation) and an inequality due to Edmundson and Madansky ([7], [25], the generalization of the trapezoidal approximation). For convex functions in ξ, Jensen provides a lower bound while Edmundson-Madansky provides an upper bound. Significant refinements of these bounds appear in Huang, Ziemba, and Ben-Tal [20], Kall and Stoyan [22] and Frauendorfer [11].

We refer to a general integrand $f(x,\boldsymbol{\xi})$, where $\boldsymbol{\xi} = \xi(\omega)$, ω has an associated probability measure P, and f represents the value of a problem such as (2.2) for a single realization ω without the nonanticipativity constraint. The true stochastic program can then be written as

$$\min_{x \text{ nonanticipative}} \int_{\Omega} f(x,\boldsymbol{\xi}) P(d\omega). \tag{3.1}$$

To construct (2.2), we need to determine an appropriate set of scenarios to replace the multiple integration in (3.1). The basic ideas are to partition the support Ξ into a number of different regions (analogous to intervals in one-dimensional integration) and to apply bounds in each of those regions. We let the partition of Ξ be $\mathcal{S}^{\nu} = \{S^l, l = 1,\ldots,\nu\}$. Define $\xi^l = E[\boldsymbol{\xi}|S^l]$ and $p^l = P[\boldsymbol{\xi} \in S^l]$. The basic lower bounding result is the following.

Theorem 1. *Suppose that $f(x,\cdot)$ is convex for all $x \in D$, then*

$$E_f(x) \geq \sum_{l=1}^{\nu} p^l f(x,\xi^l). \tag{3.2}$$

Proof: Write $E_f(x)$ as

$$\begin{aligned} E_f(x) &= \sum_{l=1}^{\nu} \int_{S^l} f(x,\boldsymbol{\xi}) P(d\boldsymbol{\xi}) \\ &= \sum_{l=1}^{\nu} p^l E[f(x,\boldsymbol{\xi})|S^l] \\ &\leq \sum_{l=1}^{\nu} p^l f(x, E[\boldsymbol{\xi}|S^l]), \end{aligned}$$

where the last inequality follows from Jensen's inequality that the expectation of a convex function of some argument is always greater than or equality to the function evaluated at the expectation of its argument, i.e., $E(f(\boldsymbol{\xi})) \geq f(E(\boldsymbol{\xi}))$. □

This result applies directly to (2.2) by letting each $\sigma = \xi^l$ and $p(\sigma) = p^l$. The approximating distribution P^ν is the discrete distribution with atoms of probability p^l at each ξ^l for $l = 1, \ldots, \nu$. By choosing $\mathcal{S}^{\nu+1}$ so that each $S^l \in \mathcal{S}^{\nu+1}$ is completely contained in some $S^{l'} \in \mathcal{S}^\nu$, the approximations actually improve, i.e.,

$$E_f(x) \geq E_f^{\nu+1}(x) \geq E_f^\nu(x). \tag{3.3}$$

Various methods can achieve convergence in distribution of the P^ν to P.

In general, the goal of refining the partition from ν to $\nu+1$ is to achieve as great an improvement as possible. We describe the basic approaches below. More details appear in Birge and Wets [3] and Frauendorfer and Kall [13].

Three basic decisions are to choose the cell, $S^{\nu^*} \in \mathcal{S}^\nu$, in which to make the partitiom, to choose the direction in which to split, S^{ν^*}, and to choose the point at which to make the split. For ease of exposition, suppose that the sets S^l are all rectangular, defined by $[a_1^l, b_1^l] \times \cdots \times [a_N^l, b_N^l]$. The most basic refinement scheme for $l = \nu^*$ then is to find i^* and $c_{i^*}^l$ so that $S^l(\nu)$ splits into $S^l(\nu+1) = [a_1^l, b_1^l] \times \cdots [a_{i^*}^l, c_{i^*}^l] \times [a_N^l, b_N^l]$ and $S^{\nu+1}(\nu+1) = [a_1^l, b_1^l] \times \cdots [c_{i^*}^l, b_{i^*}^l] \times [a_N^l, b_N^l]$.

If we also have an upper bound, $UB(S^l) \geq E[f(x,\xi)|\xi \in S_l^\nu]$ for each cell S^l, then the most likely choice for S^{ν^*} is the cell which maximizes $p_l(UB(S^l) - f(x,\xi^l))$, which bounds the error attributable to the

approximation on S^l. Reducing this greatest partition error appears to offer the most hope in reducing the error on the $\nu + 1$ approximation.

The direction choice is somewhat less clear. The general idea is to choose a direction in which the function f is "most nonlinear." The use of subgradient (dual price) information for this process was discussed in [3]. Frauendorfer and Kall [11] improved on this. The general idea is to choose the direction that yields the maximum of the minimum of linearization errors in each direction.

These refinement schemes clearly depend on having upper bounds available. These bounds are generally based on convexity properties of f and the ability to obtain each ξ in terms of the extreme points. The fundamental result is the following theorem that also appears in Birge and Wets [3].

Theorem 2. *Suppose that $\xi \mapsto f(x,\xi)$ is convex, Ξ is compact and $ext\Xi$ is the set of extreme points of $co\Xi$, the convex hull of Ξ. For all $\xi \in \Xi$, let $\nu(\xi,\cdot)$ be a probability measure on $(ext\Xi, \mathcal{E})$ where $\mathcal{E}$ is the Borel field of $ext\Xi$, such that*

$$\int_{e\in ext\Xi} e\nu(\xi, de) = \xi,$$

and $\omega \mapsto \nu(\xi(\omega), A)$ is measurable for all $A \in \mathcal{E}$. Then,

$$E_f(x) \leq \int_{e\in ext\Xi} f(x,e)\lambda(de), \tag{3.4}$$

where λ is the probability measure on $\mathcal{E}$ defined by

$$\lambda(A) = \int_\Omega \nu(\xi(\omega), A)P(d\omega).$$

Proof: Since f is convex in ξ, for ν,

$$f(x,\xi) \leq \int_{e\in ext\Xi} f(x,e)\nu(\xi, de).$$

Now, substituting $\xi(\omega)$ for ξ and integrating with respect to P the result in (3.4) is obtained. □

This result states that, if we can choose the appropriate ν and find λ, then we can produce an upper bound. The key is to make the calculation of λ as simple as possible. Of course, the cardinality of $ext\Xi$ may also play a role in the computability of the bound.

One way to reduce the cardinality of the supporting extreme points is simply to choose the extreme point that has the highest value as an upper bound. Let this upper bound be $UB^{\max}(x) = \sup_{e\in ext\Xi} f(x,e) \geq \int_{e\in ext\Xi} f(x,e)\lambda(de) \geq E_f(x)$ from Theorem 2, regardless of the particular λ. While $UB^{\max}$ may only involve

a single extreme point, it is often a poor bound. Its calculation also often involves evaluating all of the extreme points to maximize the convex function $f(x,\cdot)$.

In general, bounds built upon the result in Theorem 2 construct the probability measure λ so each extreme point, e_k, of Ξ has some weight, $p_k = \lambda(e_k)$. The following bounds, described in more detail in [3], find these weights in various cases. The first is general but involves some optimization. The second involves simplicial regions while the third uses rectangular regions.

Since λ is constructed to be consistent with the distribution of $\boldsymbol{\xi}$, we must have that

$$\begin{aligned}\int_\Omega \xi(\omega)P(d\omega) &= \int_\Omega int_{e\in ext\Xi} e\nu(\xi(\omega),de)P(d\omega)\\ &= \int_{e\in ext\Xi} e\int_\Omega \nu(\xi(\omega),de)P(d\omega)\\ &= \int_{e\in ext\Xi} e\lambda(de).\end{aligned}$$

Hence, $\lambda \in \mathcal{P} = \{\mu|\mu$ is a probability measure on $\mathcal{E}$, and $E_\mu[e] = \bar{\xi}\}$. The next upper bound, orginally suggested by Madansky [26] and extended by Gassmann and Ziemba [16], builds on this idea by finding an upper bound through a linear program to maximize the objective expectation over all probability measures in $\mathcal{P}$. We write this bound as UB^{mean}, where

$$UB^{mean}(x) = \begin{array}{lll} \max_{p_1,\ldots,p_k} & \sum_{k=1}^K p_k f(x,e_k) & \\ \text{s. t.} & \sum_{k=1}^K p_k e_k & = \bar{\xi}, \\ & \sum_{k=1}^K p_k & = 1, \\ & p_k \geq 0, & k = 1,\ldots,K. \end{array} \tag{3.5}$$

The probability measure that optimizes the linear program in (3.5) is the solution of a moment problem in which only the first moment is known. Another interpretation of this bound is that it represents the worst possible outcome if only the mean of the random variable is known. Optimizing with this bound, therefore, brings some form of risk avoidance if no other distribution information is available.

Assuming that the dimension of coΞ is N, Carathéodory's theorem states that $\bar{\xi}$ must be expressable as a convex combination of at most $N+1$ points in $ext\Xi$. Finding these $N+1$ points may however again involve computations for the values at all extreme points. The number of extreme point representations may be much higher that $N+1$ if Ξ is, for example, rectangular, but, for simplicial Ξ, the representation is, in fact, unique. Indeed the p_k in this case are the barycentric coordinates of $\bar{\xi}$.

Although Ξ may not be simplicial itself, it is often possible to extend $f(x,\cdot)$ from Ξ to some simplex Σ including Ξ. The bound obtained with this approach is written UB^Σ. In this bound, the number of points

used in the evaluation remains one more than the dimension of the affine hull of Ξ. Frauendorfer [12] gives more details about this form of approximation and various methods for its refinement.

Often Ξ is given as a rectangular region. In this case, the number of extreme points is 2^N. The number of simplices containing $\bar{\xi}$ may also be exponential in N. With relatively complete information about the correlations among random variables, however, bounds can be obtained that assign the same weight to each extreme point of Ξ (or a rectangular enclosing region), regardless of the value of x. This attribute is quite beneficial in algorithms where x may change frequently as an optimal solution is sought.

The basic bounds for rectangular regions follow Edmundson and Madansky (see [26]), for which, the name *Edmundson-Madansky (E-M) bound* is used. They begin with the trapezoidal type of approximation on an interval. Here if $\Xi = [a, b]$, then we can easily construct $\nu(\xi, \cdot)$ in Theorem 2 as $\nu(\xi, a) = \pi(\xi)$ and $\nu(\xi, b) = 1 - \pi(\xi)$, where $\pi(\xi) = \frac{b-\xi}{b-a}$. Integrating over ω, we obtain

$$\begin{aligned} \lambda(a) &= \int_\Omega \nu(\xi(\omega), A) P(d\omega) \\ &= \int_\Omega \frac{b - \xi(\omega)}{b-a} P(d\omega) \\ &= \frac{b - \bar{\xi}}{b-a}. \end{aligned} \tag{11}$$

We also then have $\lambda(b) = \frac{\bar{\xi}-a}{b-a}$. The bound obtained is $UB^{EM}(x) = \lambda(a)f(x,a) + \lambda(b)f(x,b) \geq E_f(x)$. This bound represents approximating the integrand $f(x, \cdot)$ with the values formed as convex combinations of extreme point values. This is the same procedure as in trapezoidal approximation for numerical integration except that the endpoint weights may change for nonuniform probability distributions.

The E-M bound on an interval extends easily to multiple dimensions, where $\Xi = [a_1, b_1] \times \cdots \times [a_N, b_N]$ if either $f(x, \cdot)$ is separable in the components of ξ, in which case, the bound is just applied in each component separably, or the components of ξ are stochastically independent. In this case, the bound is just developed in each component $i = 1$ to N in order so the full independent ξ_i bound contains the product of all combinations of each interval bound, i.e.,

$$UB^{EM-I}(x) = \sum_{e \in ext\Xi} (\Pi_{i=1}^N \frac{|\bar{\xi}_i - e_i|}{b_i - a_i}) f(x, e),$$

where Ξ is again assumed polyhedral.

When the components of $\boldsymbol{\xi}$ are correlated, the bound is still tractable (see [11]), although somewhat more difficult to evaluate. In this case, define:

$$\delta(e, \xi_i) = \begin{cases} (\xi_i - a_i) & \text{if } e_i = a_i, \\ (b_i - \xi_i) & \text{if } e_i = b_i. \end{cases}$$

Then, we have that

$$\nu(\xi, e) = \Pi_{i=1}^{N} \frac{\delta(e, \xi_i)}{(b_i - a_i)}.$$

The $\lambda(e)$ values can be found by integrating over ω. This may involving all products of the ξ_i components. Defining $\mathcal{M} = \{M | M \subset \{1, \ldots, N\}\}$, and $\rho_M = E[\Pi_{i \in M} \xi_i] - \Pi_{i \in M} \bar{\xi}_i$, we obtain the general E-M extension:

$$UB^{EM-D}(x) = UB^{EM-I}(x) + \sum_{e \in ext \Xi} \frac{1}{\Pi_{i=1}^{N}(b_i - a_i)} \{ \sum_{M \in \mathcal{M}} [\Pi_{i \notin M} (-1)^{\frac{e_i - a_i}{b_i - a_i}} (a_i (\frac{e_i - a_i}{b_i - a_i}) + b_i (\frac{b_i - e_i}{b_i - a_i})) \times \Pi_{i \in M} (-1)^{1 - \frac{e_i - a_i}{b_i - a_i}}] \rho_M \} f(x, e). \tag{3.6}$$

Notice, in (3.6), that if the components of ξ are independent, then $\rho_M = 0$ for all M and $UB^{EM-D}(x) = UB^{EM-I}(x)$ as expected.

Each of these upper bounds is a solution of a corresponding moment problem in which the highest expected function value is found over all probability distributions with the given moment information. The upper bounds derived here used first moment information plus some information about correlations. Other bounds with higher moments are also possible.

For different support regions, Ξ, we can combine the above bounds or use enclosing regions as we mentioned in terms of simplicial approximation. For applying the bounds in a convergent method, the partitioning scheme in Theorem 1 is again applied. Instead of applying the bounds on Ξ in its entirety, they are applied on each S^l. The dimension of these cells may, however, make computations quite cumbersome, especially if the S^l have exponential numbers of extreme points. For this reason, algorithms primarily concentrate on a lower bounding approximation for most computations and only use the upper bound to check optimality and stopping conditions.

So far, we only considered convex $f(x, \cdot)$. In (2.2), the value function may be convex in random constraint parameters but not if objective coefficients are also random. In this general case, the Jensen type bounds can provide an upper bound while the extreme point bounds can provide lower bounds. We can combine these results with the convex function results to obtain overall bounds with relatively straighforward computations as in (3.6) (see [10]) or extensions of the approach in UB^{mean} (Edirisinghe [6]).

Given the realizations represented in these bounds, the next step in stochastic programming is to solve the resulting problem. The main procedures are based on large-scale mathematical programming procedures constructed to take advantage of the structure of stochastic programs. In the next section, we describe the main type of methods for multi-stage stochastic programs such as the financial planning problem in Section 2.

4. Nested Decomposition Procedures for Multistage Stochastic Linear Programs

Nested decomposition procedures were proposed for deterministic models by Ho and Manne [19] as well as Glassey [17]. These approaches are essentially inner linearizations that begin treat all previous periods as subproblems to a current period master problem. The previous periods generate columns that can be used by the present period master problem.

A difficulty with these primal nested decomposition or inner linearization methods is that the set of inputs may be fundamentally different for different last period realizations. Since the number of last period realizations is the total number of scenarios in the problem, these procedures are not well adapted to efficient implementations. Some success has been achieved, however, by Noël and Smeers [29].

Instead of inner linearization, the general approach has been to use the outer linearization or generalization of the two-stage L-shaped method. Louveaux [24] first performed this generalization for multistage quadratic problemsw. Birge [2] extended the two-stage method in the linear case as in the following description.

The basic idea of the nested L-shaped (see [33]) or Benders' decomposition method is to decompose a problem in the form of (2.1) into distinct subproblems for each stage and scenario. They are linked by cuts on future objective values as a function of a current decision x^t. Other cuts achieve an x^t that has a feasible completion in all descendant scenarios. The cuts represent successive linear approximations of the future objective including feasibility. If the future objective function is polyhedral, this process converges to an optimal solution in a finite number of steps.

In general, for every stage $t = 1, \ldots, T-1$, and each scenario at that stage, $k = 1, \ldots, K^t (= N^1 \cdots N^t)$, we have the following master problem that generates cuts to stage t and proposals for stage $t+1$:

$$\min \; c_k^{tT} x_k^t + \theta_k^t \tag{4.1}$$

$$\text{s. t. } W^t x_k^t = h_k^t - T_k^{t-1} x_{a(k)}^{t-1} \tag{4.2}$$

$$D_{k,j}^t x_k^t \geq d_{k,j}^t, j = 1, \ldots, r_{k,j}^t, \tag{4.3}$$

$$E_{k,j}^t x_k^t + \theta_k^t \geq e_{k,j}^t, j = 1, \ldots, s_{k,j}^t, \tag{4.4}$$

$$x_k^t \geq 0, \tag{4.5}$$

where $a(k)$ is the ancestor scenario of k at stage t, $x_{a(k)}^{t-1}$ is the current solution from the previous period, and where for $t = 1$, we interpret $b = h^1 - T^0 x^0$ as the initial conditions of the problem. We may refer also to the stage T problem in which θ_k^T and constraints (4.3) and (4.4) are not present. We will describe a basic algorithm for iterating among these stages. We then discuss some enhancements of this basic approach. In

the following, $\mathcal{D}^t(j)$, denotes the period t descendants of a scenario j at period $t-1$. We assume that all variables have finite upper bounds to avoid complications presented by unbounded solutions (although again these can be treated as in Van Slyke and Wets [33]).

Nested Benders' Decomposition Method for Multistage Stochastic Linear Programs

Step 0. Set $t = 1$, $k = 1$, $r_k^t = s_k^t = 0$, add the constraint $\theta_k^t = 0$ to (4.1-4.5) for all t and k, and let $DIR = FORE$. Go to 1.

Step 1. Solve the current problem (1-5) for t and k. If infeasible and $t = 1$, then stop, problem (2.1) is infeasible. If infeasible and $t > 1$, then, let $r_{a(k)}^{t-1} = r_{a(k)}^{t-1} + 1$, let $DIR = BACK$. Let the infeasibility condition obtained be $\pi_k^t, \rho_k^t \geq 0$ such that $\pi_k^{tT} W^t + \rho_k^{tT} D_k^t \leq 0$ but $\pi_k^{tT}(h_k^t - T_k^{t-1} x_{a(k)}^{t-1} + \rho_k^{tT} d_k^t > 0$. Let $D^{t-1}_{a(k),r^{t-1}_{a(k)}} = \pi_k^{tT} T_k^{t-1}$, $d^{t-1}_{a(k),r^{t-1}_{a(k)}} = \pi_k^{tT} h_k^t + \rho_k^{tT} d_k^t$. Let $t = t-1$, $k = a(k)$ and return to (1).
If feasible, update the values of x_k^t, θ_k^t, and store the value of the dual multipliers on constraints (4.2-4.4) as $(\pi_k^t, \rho_k^t, \sigma_k^t)$, respectively. If $k < K_t$, let $k = k+1$, and return to (1). Otherwise, ($k = K_t$), if $DIR = FORE$ and $t < T$, let $t = t+1$ and return. If $t = T$, let $DIR = BACK$. Go to 3.

Step 2. For all scenarios $j = 1, \ldots, K^{t-1}$ at $t-1$, compute

$$E_j^{t-1} = \sum_{k \in \mathcal{D}^t(j)} \frac{p_k^t}{p_j^{t-1}} \pi_k^t T_k^t$$

and

$$e_j^{t-1} = \sum_{k \in \mathcal{D}^t(j)} \frac{p_k^t}{p_j^{t-1}} [\pi_k^t h_k^t + \sum_{i=1}^{r_k} \rho_{ki}^t d_{ki}^t + \sum_{i=1}^{s_k} \sigma_{ki}^t e_{ki}^t].$$

The current conditional expected value of all scenario problems in $\mathcal{D}^t(j)$ is then $\bar{\theta}_j^{t-1} = e_j^{t-1} - E_j^{t-1} x_j^{t-1}$. If the constraint $\theta_j^{t-1} = 0$ appears in (4.1-4.5) for $t-1$ and j, then remove it, let $s_j^{t-1} = 1$, and add a constraint (4.4) with E_j^{t-1} and e_j^{t-1} to (4.1-4.5) for $t-1$ and j.

If $\bar{\theta}_j^{t-1} > \theta_j^t$, then let $s_j^{t-1} = s_j^{t-1} + 1$, and add a constraint (4.4) with E_j^{t-1} and e_j^{t-1} to (4.1-4.5) for $t-1$ and j. If $t = 2$ and no constraints are added to (4.1-4.5) for $t-1 = 1$ ($j = K^1 = 1$,), then stop with x_1^1 optimal. Otherwise, let $t = t-1$, $j = 1$. If $t = 1$, let $DIR = FORE$. Go to 1.

Many alternative strategies are possible in this algorithm in terms of determining the next subproblem (4.1-4.5) to solve. For feasible solutions, the description above explores all scenarios at t before deciding to move to $t-1$ or $t+1$. For feasible iterations, the algorithm above proceeds from t in the direction of DIR

until it cannot proceed further in that direction. This is the "fast forward-fast back" procedure proposed by Wittrock [35] for deterministic problems and implemented with success by Gassmann [15] for stochastic problems. One may alternatively enforce a move from t to $t-1$ ("fastback") or from t to $t+1$ ("fastforward") whenever it is possible. From experiments conducted by Gassmann [15], the above fast-forward-fast-back sequencing protocol seems generally to work better than the either of these alternatives.

For infeasible solutions at some stage, this algorithm immediately returns to the ancestor problem to see whether a feasible solution can be generated. This alternative appears practical since subsequent iterations with a currently infeasible solution do not seem worthwhile.

We note that much of this algorithm can also run in parallel. We refer to Ruszczyński [32] who describes parallel procedures in detail. One should pay attention in parallel implementations to the possible additional work for solving similar subproblems. The convergence of this method is relatively straightforward (see [2]).

The main advantage of this decomposition procedure over standard mathematical programming techniques is that several subproblems (4.1-4.5) may be quite similar for different realizations of the random variable. After one of these is solved, the others may be solved quite quickly. The basic procedures for doing this efficiently are called *sifting* (see Gartska and Rutenberg [14]) and *bunching* (see Wets [34] and Gassmann [15]). They generally avoid repeated matrix operations while only saving the set of subproblems solved by a given basis. The result is often a dramatic decrease in solution time in comparison to non-specialized techniques. For example, Birge [2] reports speed-ups from 10 to 75 in comparison to MINOS (Murtagh and Saunders [28]).

Other techniques have also been suggested based for stochastic programming. Some use sampling techniques while others focus on relaxing the nonanticipativity constraints. In sampling procedures, the main methods are stochastic quasi-gradient methods (see Ermoliev [8]) plus methods based on outer linearizations such as stochastic decomposition (Higle and Sen [18]) and importance sampling (Dantzig and Glynn [4]).

The procedures which relax nonanticipativity place these constraints into the objective with a lagrangian term. One method called progressive hedging (Rockafellar and Wets [31]) use the proximal point algorithm (see Rockafellar [30]) to obtain a lagrangian saddle point. The other procedure (see Dempster [5]) uses dual ascent steps in an augmented lagrangian framework. Both of these procedures are especially advantageous when the original problem has a special structure (such as a network see Mulvey and Vladimirou [27]) that would be destroyed by adding the nonanticipitavity constraints.

5. Conclusions

This paper presented a brief discussion of stochastic programming methods and models. We began with a simple example in financial planning that illustrated the value of the stochastic solution over that of a deterministic model solution. We also showed how this quantity is different from the expected value of perfect information.

We then described the main procedures for creating finite dimensional problems when the original problem has an infinite range of possibilities. We showed how these approximations could be used to obtain bounds on the optimal objective value.

In the last section, we described the most common solution procedure based on outer linearizations of the value function. This method obtains significant computational advantages over non-specialized procedures by its exploiting the similarities among problems only differing in some random parameters. Overall, these methods allow large and practical stochastic programs, and, with continued advances in computational capabilities, should lead to widespread solutions of models that explicitly incorporated randomness.

References

1. J. Birge, "The value of the stochastic solution in stochastic linear programs with fixed recourse," **Mathematical Programming** 24 (1982) 314-325.
2. J. Birge, "Decomposition and partitioning methods for multi–stage stochastic linear programs," **Operations Research** 33 (1985) 989-1007.
3. J.R. Birge and R. J–B. Wets," Designing approximation schemes for stochastic optimization problems, in particular, for stochastic programs with recourse," **Math. Progr. Study** 27 (1986) 54-102.
4. G. Dantzig and P. Glynn, "Parallel processors for planning under uncertainty,"**Annals of Operations Research** 22 (1990) 1-21.
5. M.A.H. Dempster, "On stochastic programming II: dynamic problems under risk," **Stochastics** 25 (1988) 15-42.
6. N.C.P. Edirisinghe, "Essays on Bounding Stochastic Programming Problems," Ph.D. dissertation, The University of British Columbia, 1991.
7. H.P. Edmundson, "Bounds on the expectation of a convex function af a random variable," RAND Corporation Paper 982, Santa Monica, CA, 1956.

8. Y. Ermoliev, "Stochastic quasigradient methods and their applications to systems optimization," **Stochastics** 9 (1983), 1–36.

9. R. Fourer, D. M. Gay, and B. W. Kernighan, "AMPL: A mathem atical Programming Language, " Computing Science Technical Report No. 133, AT& T Bell Laboratories, Murray, Hill, NJ, 1987.

10. K. Frauendorfer,"Solving SLP recourse problems: the case of stochastic technology matrix, RHS, and objective," in **Proceedings of 13th IFIP Conference on System Modelling and Optimization**, Springer-Verlag, Berlin, 1988.

11. K. Frauendorfer, Solving S.L.P. Recourse Problems with Arbitrary Multivariate Distributions – the Dependent Case, **Mathematics of Operations Research** 13 (1988) 377-394.

12. K. Frauendorfer, "A simplicial approximation scheme for convex two-stage stochastic programming problems," Manuskripte, Institut für Operations Research, University of Zurich, Zurich, July 1989.

13. K. Frauendorfer and P. Kall, "A solution method for SLP recourse problems with arbitrary multivariate distributions - the independent case," **Problems Control Inf. Theory** 17 (1988) 177-205.

14. S. Gartska and D. Rutenberg, "Computation in discrete stochastic programs with recourse," **Operations Research** 21 (1973) 112-122.

15. H. I. Gassmann, " MSLiP: A computer code for the multistage stochastic linear porgramming problem," **Mathematical Programming** 47 (1990) 407-423.

16. H. I. Gassmann and W. T. Ziemba, "A tight upper bound for the expectation of a convex function of a multivariate random variable," **Math. Progr. Study** 27 (1986) 39-53.

17. C.R. Glassey, "Nested decomposition and multistage linear pprograms," **Mgmt. Sci.** 20 (1973) 282-292.

18. J. Higle and S. Sen, "Stochastic decomposition: an algorithm for two stage linear programs with recourse,", **Math. of O.R.** 16 (1991) 650-669.

19. J.K. Ho and A.S. Manne, "Nested decomposition for dynamic models," **Math. Program.** 6 (1974) 121-140.

20. C.C. Huang, W. Ziemba, and A. Ben-Tal," Bounds on the Expectation of a Convex Function of a Random Variable: with Applications to Stochastic Programming,"**Operations Research** 25 (1979) 315–325.

21. J.L. Jensen, Sur les fonctions convexe et les inégalités entre les valeurs moyennes, **Acta. Math.** 30 (1906) 175-193.

22. P. Kall and D. Stoyan, "Solving stochastic programming problems with recourse including error bounds," **Math. Operationsforsch. Statist. Ser. Optim.** 13 (1982) 431-447.

23. J. G. Kallberg and W.T. Ziemba, "Comparison of alternative utility function sin protfolio selection problems, **Management Science** 29 (1983) 1257-1276.

24. F. V. Louveaux, "A solution method for multistage stochastic programs with recourse with application to an energy investiment problem," **Operations Research** 28 (1980) 889-902.

25. A. Madansky, "Bounds on the expectation of a convex function of a multivariate random variable," **Ann. Math. Statist.** 30 (1959) 743-746.

26. A. Madansky, "Inequalities for stochastic linear programming problems," **Management Science** 6 (1960) 197-204.

27. J. Mulvey and H. Vladimirou, "Solving multistage stochastic networks: an application of scenario aggregation," Dept. of CE & OR, Princeton University, SOR 88-1, 1988.

28. B.A. Murtagh and M.A. Saunders, "MINOS 5.0 User's Guide," Technical Report SOL 83-20, Systems Optimization Laboratory, Stanford University, Stanford, CA, 1983.

29. M.-C. Noël and Y. Smeers, "Nested decoomposition of multistage nonlinear programs with recourse," **Math. Program.** 37 (1987) 131-152.

30. R. T. Rockafellar, "Monotone operators and the proximal point algorith," **SIAM J. Control Opt.** 14 (1976) 877-898.

31. R.T. Rockafellar and R. J-B. Wets, "Scenarios and policy aggregation in optimization under uncertainty," **Mathematics of Operations Research** 16 (1991) 119-147.

32. A. Ruszczyński, "A regularized decomposition for minimizing a sum of polyhedral functions," **Mathematical Programming** 35 (1986) 309-333.

33. R. Van Slyke and R. J-B. Wets, "L-shaped linear programs with application to optimal control and stochastic programming," **SIAM Journal on Applied Mathematics** 17 (1969) 638–663.

34. R. Wets, "Large-scale linear programming techniques in stochastic programming," in: Y. Ermoliev, R. Wets, eds., **Numerical Techniques for Stochastic Optimization**, Springer, Berlin, 1988.

35. R.J. Wittrock, "Advances in a nested decomposition algorithm for solving staircase linear programs," Technical Report SOL 83-2, Systems Optimization Laboratory, Stanford University, Stanford, CA, 1983.

36. W. T. Ziemba and R.G. Vickson, eds., **Stochastic Optimization Models in Finance**, Academic Press, New York, 1975.