SCENARIO OPTIMIZATION

Ron S. DEMBO

Algorithmics Inc., 489 College Street, Suite 301, Toronto, Ontario, Canada M6G 1A5

Uncertainty in the parameters of a mathematical program may present a modeller with considerable difficulties. Most approaches in the stochastic programming literature place an apparent heavy data and computational burden on the user and as such are often intractable. Moreover, the models themselves are difficult to understand. This probably explains why one seldom sees a fundamentally stochastic model being solved using stochastic programming techniques. Instead, it is common practice to solve a deterministic model with different assumed scenarios for the random coefficients. In this paper we present a simple approach to solving a stochastic model, based on a particular method for combining such scenario solutions into a single, feasible policy. The approach is computationally simple and easy to understand. Because of its generality, it can handle multiple competing objectives, complex stochastic constraints and may be applied in contexts other than optimization. To illustrate our model, we consider two distinct, important applications: the optimal management of a hydro-thermal generating system and an application taken from portfolio optimization.

Keywords: Optimization under uncertainty, stochastic programming.

1. Introduction

Consider linear optimization problems of the form:

$$\begin{aligned} \text{Minimize} \quad & c^{\mathrm{T}}x \\ \text{subject to} \quad & Ax = b, \\ & x \geq 0. \end{aligned}$$

In many situations a large portion of the data may be uncertain, that is dependent on future events. To represent this, we further refine the original representation as the following *stochastic linear program:*

$$\begin{aligned} \text{Minimize} \quad & c_u^{\mathrm{T}}x && (\text{"uncertain objective"}) \\ \text{subject to} \quad & A_u x = b_u && (\text{"uncertain constraints"}) \\ & A_d x = b_d && (\text{"deterministic constraints"}) \\ & x \geq 0. \end{aligned}$$

An extremely powerful, convenient and natural way to represent uncertainty is to use scenarios. For the purposes of this paper, we define a scenario as a particular realization of the uncertain data, c_u, A_u and b_u, represented by c_s, A_s and b_s. Thus, for each scenario $s \in S \equiv \{\text{set of all scenarios}\}$, the above problem reduces to the deterministic problem below, which we will refer to as the *scenario subproblem:*

$$
\begin{aligned}
v_s \equiv \quad & \text{Minimize} \quad && c_s^{\mathrm{T}} x \\
& \text{subject to} && A_s x = b_s, \\
& && A_d x = b_d, \\
& && x \geqslant 0.
\end{aligned}
$$

Let x_s denote the solution to this problem.

Associated with each scenario is a probability p_s. For the moment we assume that p_s, $s \in S$ are given. It is here where our scenario approach differs from the more classical approaches in the stochastic programming literature. In a dynamic situation *we will assume that these probability weights are changing over time*. In certain instances the exact nature of the underlaying stochastic process may be known. More often than not however, it will not and we will expect periodically to revise any policy (i.e. solution). In the stochastic programming literature it is more typical for one to assume that all uncertain parameters are known a priori in terms of some distribution or stochastic processes. This information is then used to take a decision here and now that is in some sense valid for all future time. We prefer to assume a priori that we will have to revise any policy suggested today in the light of *future changes to our probability estimates*. This implies a rolling horizon or control framework in which we monitor the probability estimates and revise our policy (re-solve the coordinating model) when significant changes occur.

Whereas it is clear that the solution of a single scenario problem poses no difficulty, solving such problems still does not provide a definite way of determining what a reasonable solution to the original stochastic problem should be. The fundamental question in scenario analysis is: "*Just how should the solutions under a number of different scenarios be combined to form a single reasonable solution to the underlying stochastic problem*?"

To attempt to answer this question we first step back and define a what we consider to be a *reasonable* or *feasible* solution to a stochastic linear system.

DEFINITION

A solution $\hat{x}$ to a stochastic linear system

$$
\begin{aligned}
& A_s x = b_s, \quad s \in S; \\
& A_d x = b_d; \\
& x \geqslant 0
\end{aligned}
$$

is said to be (*norm*) *feasible* if it satisfies the deterministic constraints $A_d \hat{x} = b_d$, $\hat{x} \geqslant 0$ and minimizes $\Sigma_s p_s \| A_s x - b_s \|$.

This definition is constructive in that it indicates a reasonable way in which scenario solutions may be combined into a single "feasible" solution to the underlying stochastic programming problem. To see how this definition applies to the problem at hand, recall that an optimal solution to a single scenario can be expressed as a solution to a stochastic linear system simply by including the equality $c_s^{\mathrm{T}}x = v_s$. For example, one possible coordination model could be:

$$\begin{aligned} \text{Minimize} \quad & \sum_s p_s \| c_s^{\mathrm{T}}x - v_s \|^2 + \sum_s p_s \| A_s x - b_s \|^2 \\ \text{subject to} \quad & A_d x = b_d, \\ & x \geqslant 0. \end{aligned}$$

In this example, the coordinating model *tracks* the scenario solutions as closely as possible while still maintaining feasibility, although not strictly in the sense of the above definition (i.e. we are minimizing the norm squared and not the norm itself). For this reason and for reasons that will become clear in our examples, we will sometimes refer to this model as a *Tracking Model.*

The coordinating model is, however, every flexible in that it may include many additional tracking terms or constraints. In fact, there is no reason why it could not include terms which attempt to track any given function of the individual scenario solutions. In sections 2 and 4 we give some examples of just how flexible the coordinating model may be.

To summarize, the *Scenario Optimization* approach to stochastic programming proceeds as follows.

Stage 1: Compute a solution to the (deterministic) problem under all scenarios.
Stage 2: Solve a coordinating or tracking model to find a single, feasible policy.

The problem referred to in stage 1 may be a linear, nonlinear or even integer programming problem. It could also be a system of equations with stochastic coefficients or any function dependent on stochastic parameters. Since, by definition, for any assumed scenario this problem is deterministic, in principle we can find a solution using known algorithms.

Stage 1 may be viewed as a sampling of the solution space of the underlying stochastic model. Stage 2 attempts to find a single "feasible" policy that best "fits" the behavior of the system under uncertainty.

Much of the power of this approach lies in stage 2 whose precise formulation will be context dependent. The stage 1 and 2 problems, all of which are deterministic and may be solved by known methods, may or may not have the same form. In stage 2, for example, we may only wish to track some particular function of the problem variables; we may wish to add constraints not present in the stage 1 problem, etc.

The work required in such an approach is a multiple of that required for a single scenario deterministic problem. This may be less than the *Scenario Aggre-*

gation approach advocated by Rockafellar and Wets [7]. It also parallelizes in a natural way. With one processor per scenario and one for the tracking problem, one should expect the entire process to require an elapsed time of the order of a small multiple of the time required for the solution of a single deterministic stage 1 problem. This behavior has been observed in other models (see e.g. [5]) and is indeed the goal in any stochastic programming computation.

It is interesting to contrast our approach with Scenario Aggregation [7].

Rockafeller and Wets [7] propose solving a large deterministic equivalent of a stochastic model, obtained by expanding the original model with copies of itself, one for each scenario, and with nonanticipativity constraints that link the individual scenario problems. They propose a "hedging" algorithm for solving this problem via a decomposition technique. Their approach is convergent and is rich in interpretation, however, as with many decomposition algorithms its rate of convergence may be poor. It may require the solution of a large number of single scenario subproblems as well as many coordinating master problems.

The fundamental difference between the Scenario Optimization approach we propose and the above Scenario Aggregation approach devised by Rockafellar and Wets is one of modelling. The two approaches emphasize two different models of the underlying stochastic problem. Scenario Optimization assumes that the scenario probabilities evolve over time in a manner that is difficult to predict or model using a stochastic process. As such, in Scenario Optimization, we take as given that the model will have to be solved periodically to readjust a policy over time (an open-loop control framework). The model is solved for one period only and the stochastic parameters, that is the scenario probabilities, are re-evaluated before proceeding to a policy for the next period. That is, we explicitly assume that these probabilities are functions of time. The coordinating model allows us to select a policy for the immediate future, which hedges against future uncertainty, based on the best current estimates of the future scenarios and their associated probabilities. In stochastic programming terms it is therefore a two-stage model. As such, it is an approximation, similar to the two-stage model used in Kusy and Ziemba [5] and elsewhere.

Scenario Aggregation, on the other hand, handles multistage stochastic programming problems. It allows for decisions that depend on future outcomes and, hence, become *nonanticipative* in future periods [7]. Current decisions may, therefore, be based on implementable future decisions.

Some implementations of Scenario Aggregation assume that the behavior of the random variables in all future stages can be predicted well. If this is the case then it makes sense to strive today to use all this information on future events to hedge the decision made today, which is what Scenario Aggregation, applied to multistage stochastic programming, attempts to do.

In many cases, as for example in the applications discussed section 4, estimates of the distributions of the random parameters in future stages are extremely poor. It is for such situations that Scenario Optimization has been designed. For such situations, it is not clear that a multistage solution is at all warranted. The appendix to this paper amplifies this point in the case of reservoir models.

2. Alternative coordination models

Following our definition of stochastic feasibility, a solution to a stochastic linear program will be feasible if it satisfies:

$$\underset{x \geqslant 0}{\text{Minimize}} \quad \sum_s p_s \| A_s x - b_s \|$$
$$\text{subject to} \quad A_d x = b_d.$$

An interesting observation is that this is a form of *exact penalty minimization*. That is, if a feasible solution to

$$A_s x = b_s, \quad \forall x \in S;$$
$$A_d x = b_d,$$
$$x \geqslant 0$$

exists, then stochastic feasibility corresponds to the deterministic notion. Otherwise, stochastic feasibility finds a solution as close as possible to the individual scenario solutions, in a weighted norm sense.

The most basic *coordinating model* has the generic form:

$$\underset{x \geqslant 0}{\text{Minimize}} \quad \sum_s p_s \left(\| c_s^{\mathrm{T}} x - v_s \| + \| A_s x - b_s \| \right)$$
$$\text{subject to} \quad A_d x = b_d.$$

Since this is a norm minimization, and hence a nondifferentiable problem, in practise it might be sufficient and computationally cheaper to solve an approximation of the above problem such as the one suggested in section 1. Other approximations such as a piecewise quadratic function obtained by smoothing the kinks in the objective may be preferable. Such functions have been used extensively in the stochastic programming literature (see King [4] for example). In principle one could use any penalty function in place of a norm.

It is instructive to examine the various distinct coordinating models that result when different norms are chosen.

Consider use of the L_1 norm; $\| z \| \equiv \Sigma_i | z_i |$, which may yield the coordinating model:

$$\underset{x \geqslant 0}{\text{Minimize}} \quad \sum_s p_s \left(| c_s^{\mathrm{T}} x - v_s | + | A_s x - b_s | \right)$$
$$\text{subject to} \quad A_d = b_d.$$

This may be expressed equivalently as the linear program:

$$\begin{aligned} \underset{w,y,x\geqslant 0}{\text{Minimize}} \quad & \sum_s p_s\left[(w_s^+ + w_s^-) + e^{\mathrm{T}}(y_s^+ + y_s^-)\right] \\ \text{subject to} \quad & A_d x = b_d; \\ & A_s x - b_s - (y_s^+ - y_s^-) = 0 \quad \forall s \in S; \\ & c_s^{\mathrm{T}} x - v_s - (w_s^+ - w_s^-) = 0, \quad \forall s \in S, \end{aligned}$$

where $e = (1, 1, \ldots, 1)^{\mathrm{T}}$. This has the distinct flavor of a recourse model. In fact, it is a simple recourse model. The precise relationship is explored in the next section.

In principle, there is no reason to limit the nature of the coordinating model. It may be anything we wish. One reasonable requirement however is that it fit within the notion of stochastic feasibility as defined above. Beyond this, the coordinating model may have constraints that are not present in the scenario subproblems. Its objective function may consist of any function of the outputs of the scenario subproblems.

For example, if there are multiple objective functions

$$(c_s^1)^{\mathrm{T}} x,\ (c_s^2)^{\mathrm{T}} x, \ldots, (c_s^m)^{\mathrm{T}} x,$$

then they could be handled at the coordinating stage with an objective function defined as follows.

$$\underset{x\geqslant 0}{\text{Minimize}} \quad \sum_s p_s\left(|(c_s^1)^{\mathrm{T}} x - v_s^1| + |(c_s^2)^{\mathrm{T}} x - v_s^2| + \ldots + |A_s x - b_s| \right)$$

where v_s^1, $v_s^2, \ldots,$ are the values taken on by these objectives under scenario s.

More examples of tracking models are given in section 5.

A very interesting feature of Scenario Optimization is that the scenario subproblems and the coordination problem are decoupled. Thus, in a situation in which all the possible scenarios were known a priori, one could solve the scenario subproblems for once and for all. Policy would then be determined at each period by simply solving the coordination problem with the current best estimates of the scenario probabilities.

More formally, in the case of the stochastic linear program introduced in section 1, the "optimal policy" at time t, $x(t)$, may be obtained from:

$$\begin{aligned} v(t): \quad \underset{x\geqslant 0}{\text{Minimize}} \quad & \sum_s p_s(t)\left(|c_s^{\mathrm{T}} x - v_s| + |A_s x - b_s| \right) \\ \text{subject to} \quad & A_d x = b_d. \end{aligned}$$

Viewing the problem in this light, the relationship to optimal control is more apparent. The control problem being:

> Given the trajectory of scenario probabilities $p_s(t)$, $s \in S$; $t = 1, \ldots, T$, compute controls $x(t)$, $t = 1, \ldots, T$ so that $\int_0^T v(t)\, \mathrm{d}t$ is minimized.

Thus, when $p_s(t)$ is known or may be computed from some known underlying stochastic process model, for all scenarios s and over the entire control period T, then the above model defines an optimal policy over the planning horizon. Typically, $p_s(t)$ may be estimated well for $t = 1$ with only rough estimates available for later periods, in which case open loop control makes sense and we revert to the simple model discussed in section 1.

Up to now, for ease of presentation, we have presented our ideas in the context of linear optimization. However, there is nothing in Scenario Optimization nor in its implementation that requires linearity. Consider, for example, the nonlinear stochastic system:

$$f(x;\ r) \geqslant 0, \quad x \in X,$$

where $f(\cdot;\ r): \mathbb{R}^n \to \mathbb{R}^m$, $x \in \mathbb{R}^n$, and $r \in \mathbb{R}^q$ is a vector of random parameters. Assume:

r_s is the value of r under scenario $s \in S$;
S is a finite set of all possible scenarios;
$p_s(t)$ is the probability of scenario $s \in S$ at time t; and
x_s satisfies $f(x_s;\ r_s) \geqslant 0$.

A *Scenario Optimization* approach would proceed as follows:

Given r_s, $p_s(t)$, $\forall s \in S$;
(*Scenario Subproblem*) Compute x_s satisfying $f(x_s;\ r_s) \geqslant 0$, $\forall s \in S$;
(*Tracking Problem*) At time t, compute a policy $x^*(t)$ satisfying

$$v(t) = \underset{x \in X}{\text{Minimize}} \quad \sum_s p_s(t) \| f(x;\ r_s) - f(x_s;\ r_s) \|.$$

As was shown earlier, the Tracking Model, $v(t)$, may also be imbedded in an optimal control framework.

The above formulation in principle covers all forms of optimization, linear, nonlinear, discrete as well as systems of equations and inequalities. We could generalize even further. Scenario Optimization could be applied to a "black box", i.e. computer simulation model. All that is needed are the input scenarios and the values of the outputs that need to be tracked.

The tracking problems $v(t_i)$ and $v(t_j)$ differ only in that they have different objective function coefficients $p_s(t_i)$ and $p_s(t_j)$ respectively. This should offer some significant computational advantages to Scenario Optimization. For the same reason, it should be relatively cheap and easy to test the effect of changes in the scenario probabilities.

From a user's perspective, once the model has been set up and scenarios identified, Scenario Optimization reduces the immense complexity of a stochastic optimization environment to one in which the only parameters that need to be determined are the scenario probabilities. This could potentially make stochastic programming accessible to nontechnical users. In cases where these probabilities are based on personal hunches, performing "what if" analyses quickly becomes both necessary and important. For the reasons outlined above, in our framework such analysis could be relatively easy.

4. Relationship to stochastic programming with recourse

In order to make the comparison, assume that the uncertain parameters in the stochastic linear program are specified in terms of discrete scenarios and their associated probabilities, i.e. $(c_s,\ A_s,\ b_s;\ p_s)$, $s \in S$. The random linear program described in section 1, formulated as a stochastic program with recourse, would take the following form (see Nazareth [5], for example).

$$\begin{aligned} &\underset{x \geqslant 0}{\text{Minimize}} && E_s\left[c_s^{\mathrm{T}}x + P_s(x)\right] \\ &\text{subject to} && A_d x = b_d, \end{aligned}$$

where $E_s[\cdot]$ is the expectation over S, $P_s(x) \equiv \min_{y \geqslant 0}\{q^{\mathrm{T}}y \mid Wy = b_s - A_s x\}$, and W is a (fixed) recourse matrix and q a given vector specifying the cost of recourse.

A special case of this formulation is

$$P_s(x) \equiv \min_{y_s^+,\, y_s^- \geqslant 0}\left\{e^{\mathrm{T}}(y_s^+ + y_s^-) \mid y_s^+ - y_s^- = b_s - A_s x\right\},$$

which is equivalent to

$$P_s(x) = |A_s x - b_s|.$$

For this special case, observing that we may add the constant $v_s \equiv c_s^{\mathrm{T}}x_s$ to the objective function, the above expression yields:

$$\begin{aligned} &\underset{x \leqslant 0}{\text{Minimize}} && \sum_s p_s\left(c_s^{\mathrm{T}}x - v_s + |A_s x - b_s|\right) \\ &\text{subject to} && A_d x = b_d. \end{aligned}$$

This is very similar to the corresponding Scenario Optimization Tracking Model except that, in Scenario Optimization, the objective would typically have $|c_s^{\mathrm{T}}x - v_s|$ instead of $(c_s^{\mathrm{T}}x - v_s)$ as in the above Stochastic Programming with Recourse (SPR) model. Note, however, that $|c^{\mathrm{T}}x - v_s|$ could be added to $P_s(x)$ to produce equivalent problems. Since $|c_s^{\mathrm{T}}x - v_s| \geqslant (c_s^{\mathrm{T}}x - v_s)$, Scenario Optimization solutions will always provide an upper bound to the above SPR model.

An interesting observation that one can derive from the above is that, for linear problems, when one replaces $|c_s^{\mathrm{T}}x - v_s|$ with $(c_s^{\mathrm{T}}x - v_s)$ in the objective function of a tracking model, the tracking model may be formed using only the data of the problem. That is, there is no longer any need to solve the scenario subproblems! This might be useful as an approximation tool in computational implementations of Scenario Optimization. This is indeed the approach in [5].

5. Some illustrative applications

5.1. PORTFOLIO IMMUNIZATION

The portfolio immunization problem is concerned with finding the cheapest set of fixed-income securities (to simplify the presentation we assume these are bonds) whose present value over some time period is equal to the present value of a given portfolio, usually a stream of liabilities.

A deterministic formulation of this problem would have the general form:

$$
\begin{aligned}
v_s \equiv \quad & \text{Minimize} \quad c_s^{\mathrm{T}} x \\
& \text{subject to} \quad 1 \leqslant x \leqslant u; \\
& \qquad \sum_{j \in J} PV_{sj} x_j \leqslant PV_{sT},
\end{aligned}
$$

where:

PV_{sj} $\equiv$ present value of bond j under discount scenario s,
$PV_s(x)$ $\equiv \sum_{j \in J} PV_{sj} x_j$; present value of portfolio x under discount scenario s,
PV_{sT} $\equiv$ present value of liability (target) portfolio under discount scenario s,
x_j $\equiv$ amount of bond j in the optimal immunizing portfolio,
c_s $\equiv$ the current vector of bond market prices,
u_j $\equiv$ maximum units of bond j allowed,
l_j $\equiv$ minimum units of bond j allowed,
J $\equiv$ set of bonds j available for immunizing.

The uncertainty in this problem stems from the present value coefficients, PV_{sj}. These fluctuate as interest rates (and hence discount rates) change over time. The deterministic model above, which assumes a single scenario s, is very easy to compute. However, notice that since it is a knapsack problem, if the bounds u_j are large enough, the solution will contain only one bond, regardless of the scenario chosen. Moreover, the optimal immunizing bond will typically be different for different scenarios.

It is clear that such a solution is not satisfactory since we know that this is likely to track poorly if the assumed discount scenario does not occur. We expect an "optimal" solution to contain a diverse portfolio in order to hedge against the uncertain future long and short-term interest rates. Notice also that, whereas the solution to any single scenario subproblem does not appear to offer a good solution to the Immunization problem, ony may be able to solve many such problems extremely cheaply. This observation appears to make Immunization a good candidate for Scenario Optimization.

In current practise, in order to improve on this situation, additional constraints are added to the deterministic immunization model and arbitrary bounds are placed on the variables to guarantee a diversified portfolio of bonds in the solution. Prescribing a solution, however, cannot be considered a good modelling practice.

In contrast, the stochastic model as we show below, in a Scenario Optimization framework, is quite simple and naturally produces a diverse porfolio. The resulting solution is likely to track the present value of the liabilities over time, without need for significant rebalancing, under many possible realizations of the future discount scenarios.

5.2. THE SCENARIO OPTIMIZATION APPROACH TO PORTFOLIO IMMUNIZATION [3]

The Scenario Optimization approach calls for the solution of the deterministic subproblems under each possible scenario. For reasonable choices of the bounds l_j and u_j the scenario subproblem solution, x_s, will always satisfy:

$$PV_s(x) = PV_{sT}.$$

Thus, one possible tracking or coordinating model could be

$$\begin{array}{ll} \underset{x}{\text{Minimize}} & \sum_{s \in S} p_s\left[\left(c_s^{\mathrm{T}}x - v_s\right)^2 + \left(PV_s(x) - PV_{sT}\right)^2\right] \\ \text{subject to} & l \leqslant x \leqslant u, \end{array}$$

where v_s is the optimal portfolio cost under scenario s and p_s is the probability of scenario s occurring.

If the cost of constructing the immunizing portfolio was an issue, we could enhance the model further by including a constraint that limits cost.

Let C be the total budget available for constructing the immunizing portfolio. A tracking model of great interest would then be the following parametric quadratic programming problem.

$$\begin{array}{lll} Q(C) \equiv & \underset{x}{\text{Minimize}} & \sum_{s \in S} P_s\left[\left(c_s^{\mathrm{T}}x - v_s\right)^2 + \left(PV_s(x) - PV_{sT}\right)^2\right] \\ & \text{subject to} & l \leqslant x \leqslant u, \\ & & \sum_{j \in J} c_j x_j \leqslant C. \end{array}$$

In this model one can explicitly examine the tradeoff between the cost of the immunization and its "quality", $Q(C)$, as measured by the error in tracking the scenario solutions. This cost vs. "risk exposure" tradeoff is extremely useful in the commercial applications of portfolio immunization.

An alternative tracking model based on using an L_1 norm in the tracking model is:

$$\begin{array}{ll} \underset{w^+, w^-, y^+, y^-, x}{\text{Minimize}} & \sum_s P_s\left[\left(y_s^+ + y_s^-\right) + \left(w_s^+ + w_s^-\right)\right] \\ \text{subject to} & PV_s(x) - \left(y_s^+ - y_s^-\right) = PV_{sT}, \quad \forall s \in S, \\ & c_s^{\mathrm{T}}x - \left(w_s^+ - w_s^-\right) = v_s, \quad \forall s \in S, \\ & l \leqslant x \leqslant u. \end{array}$$

Here, y_s^+ and y_s^- may be interpreted as the positive and negative absolute deviations from tracking the present value of the target portfolio under the s-th scenario. Naturally, this model could also be extended to a parametric model measuring the cost–quality tradeoff.

The above tracking models could also be extended to handle multiple criteria as follows.

Assume that in addition to present value we were interested in the total return under scenario s, $TR_s(x)$, on the portfolio as given by

$$TR_s(x) \equiv \sum_{j \in J} PV_{sj} Y_{sj} x_j,$$

where Y_{sj} is the average yield of bond j under scenario s.

Let TR_{sT} be the total return of the target liability portfolio under the s-th scenario. A reasonable tracking model might be:

$$\begin{aligned} V(C) \equiv \quad \underset{x}{\text{Minimize}} \quad & \sum_{s \in S} p_s \| c_s^{\mathrm{T}} x - v_s \| + \\ & \sum_{s \in S} p_s \| PV_s(x) - PV_{sT} \| + \\ & \sum_{s \in S} p_s \| TR_s(x) - TR_{sT} \| \\ \text{subject to} \quad & l \leqslant x \leqslant u, \\ & \sum_{j \in J} c_j x_j \leqslant C. \end{aligned}$$

The choice of norm used in the tracking model is context dependent. For example, if one is attempting to track an index or synthesize a security, minimizing the absolute value of the tracking error seems appropriate. In contrast, often one is only concerned with downside errors. In such situations the one sided norm $|z|_-$ would be appropriate, i.e.

$$|z|_- = \begin{cases} -z & \text{if } z < 0 \\ 0 & \text{otherwise.} \end{cases}$$

A more detailed discussion of the choice of norm may be found in [3].

The possibilities for further enhancements to the tracking model are virtually limitless. What is important to notice, however, is that the tracking problem may be moulded to suite the the context of the application. More details may be found in Dembo [3].

5.3. HYDROELECTRIC POWER SCHEDULING AND RESERVOIR PLANNING MODELS

The optimal management of reservoirs is an important problem faced by any hydroelectric power utility or any large governmental water authority. In terms of modelling, it is an application area in which the random nature of reservoir inflows plays a crucial role in determining policy. Primary sources of uncertainty in such problems are the quantity and timing of future water inflows into the reservoir system and, in the case of hydroelectric applications, the demand for electricity.

In applications involving the generation of hydroelectric power, one seeks to determine reservoir levels and releases over time that maximize the benefit of using hydro to replace costly thermal generation. Two recent examples of very different ways of modelling such systems are given in da Silva et al. [8], and Dembo et al. [2]. In [8], the problem is modelled in a Stochastic Dynamic Programming (SDP) framework whereas in [2] a Scenario Optimization (SOPT) methodology is used. Both treat the stochasticity of the problem. However, they are fundamentally different. In the appendix to this paper we contrast the two methodologies in this setting.

An example of a long-term hydroelectric scheduling model taken from [2] is given below.

$$\underset{V,R,S}{\text{Maximize:}} \quad \sum_{t=1}^{T} \sum_{j=1}^{J} B_{tj}(V_{t-1,j}, V_{tj}, R_{tj})$$

$$\text{subject to} \quad V_{tj} - V_{t-1,j} - \sum_{k \in K_j} (R_{tk} + S_{tk}) + R_{tj} + S_{tj} = I_{tj},$$

$$\underline{V}_{tj} \leqslant V_{tj} \leqslant \overline{V}_{tj},$$

$$\underline{R}_{tj} \leqslant R_{tj} \leqslant \overline{R}_{tj},$$

$$0 \leqslant S_{tj} \quad \text{for all} \quad t = 1, \ldots, T \text{ and } j = 1, \ldots, J,$$

where

$B(\cdot)$ is a stochastic nonlinear function measuring the benefit of hydro vs. thermal generation;

V_{tj} is the volume of reservoir j at the end of period t; $\overline{V}$ and $\underline{V}$ are given upper and lower limits on V;

I_{tj} is the net (stochastic) inflow to reservoir j in period t; that is, the difference between the inflow and water extracted for irrigation;

R_{tj} is the release (for generation) from reservoir j in period t; $\overline{R}$ and $\underline{R}$ are upper and lower limits on R;

S_{tj} is the amount spilt from reservoir j in period t; strictly speaking $S \geqslant 0$ only when V is at its upper bound; and

K_j is the set of reservoirs immediately upstream from reservoir j.

Randomness in the model enters in two ways: Net inflows I_{tj} are not known with certainty, especially in future periods; the benefit function $B(\cdot)$ is constructed using a least squares fit to a set of data generated by simulating thermal energy costs using forecasts of the electricty demand (see [2]). Both these forms for stochasticity are well handled by scenario optimization.

This application is a good example of a case in which, even if accurate estimates of the historical distributions of inflows are available, it is difficult to conceive of a model that will produce a single policy that will be valid over a long period. Instead, it is more reasonable to assume various scenarios (based perhaps on historical data) are known together with estimates of the probability of each of these scenarios. In terms of our model, a single scenario would be defined as a set of inflows into the reservoirs coupled with the electricity demand level for each time period in the model (the "right hand side" and objective function, respectively, of our optimization model).

It is also natural, in this context, that as time goes by we will modify our views on the probability of these estimates. However, it is important for the decisions taken today to hedge against potential alternative inflow and demand scenarios. Thus, in such applications a reasonable approach might be to solve a stochastic scenario optimization problem in a rolling horizon manner, modifying the probability of various inflow and demand scenarios as, for example, weather forcasts are updated.

So far we have ignored the issue of just how, in practice, scenarios should be generated.

This is naturally an extremely important component of Scenario Optimization. It is also one area in which much future research is needed. For the purposes of this paper we will show examples of how scenarios are generated in the two very different practical applications described above.

5.4. SCENARIO GENERATION FOR PORTFOLIO IMMUNIZATION

For portfolio immunization the constraint coefficients, PV_{sj}, are obtained by discounting cash flows according to the s-th discount function scenario. The discount function would typically be obtained from a yield curve (bond yield vs. maturity) for the bond under consideration. Thus it appears as if scenarios may be generated by generating expected yield curves. There are a number of existing models for generating yield curve scenarios (see [1] for example) and this is currently an active area of research in Finance. Almost all models are based on a "no arbitrage" condition, which, although beyond the scope of this paper, is important to mention since it is a well accepted notion and should probably be respected. Most importantly, it places constraints on the scenario generation process. We expect that in any real-world setting such constraints will exist. Another example of this is described below in the case of reservoir management.

One of these theoretical models could be used to generate scenarios. However, often in practise, yield curve scenarios may be generated purely by intuition since all we are doing in Scenario Optimization is enabling a user to hedge against the chosen scenarios and consequently any scenarios that may be "close to" the chosen ones. In our view, whether the scenarios thus chosen are realistic or not does not matter. They permit an expert to quantify the effect of hedging against a personal subjective viewpoint. Often in the world of finance, this can be as or more valuable than using an artificially generated set of scenarios based on some imperfect analytical model.

A more interesting and richer view of scenarios in this context would be to make the tuple (time, yield-curve) a scenario. In this way one could select different points in time and different expected yield curves associated with these points together with their associated probabilities of occurrence. This would result in a Tracking Model that would be able to hedge selectively at different points in time. These points in time could, for example, be chosen as the natural points at which the portfolio is to be reviewed. The scenarios could reflect hunches regarding the expected evolution over time of the yield curve.

Scenario Optimization is not all science. It does not specify a formulation that should be solved to get a solution to the underlying stochastic problem, as do most other methods. It is a framework in which much room is still left to the modeller. The art of choosing scenarios and the flexibility permitted by the Tracking Model leave much room for expert judgement.

5.5. SCENARIO GENERATION FOR RESERVOIR MODELS

To give an example of how one might generate scenarios in this context, we draw on the hydroscheduling model described in [2]. There are possibly many other reasonable ways in which one could generate scenarios for such models. There is no claim made here that the one we describe is the only method.

At the heart of any reservoir management system is a hydrological forecasting model, usually fed by an extensive relational database of historic hydrological data. In [2], the database contains 20 years of information on:

- individual hydro plants, reservoirs and river basins;
- topological data on the aquifer feeding the reservoirs;
- level/volume surface tables;
- historical flood data; and
- daily data on turbine operation, spillage, pumping etc. for each hydro plant.

Short-term planning models use forecasts to generate inflow scenarios. In the long-term models inflow scenarios are generated based on historical data. An inflow scenario is a time series of inflows for a particular reservoir system, corresponding to the duration and periods of the relevant study. For example, in the long-term model, an inflow scenario could be set of weekly inflows into the reservoirs over the year.

Each inflow scenario is artificially constructed from the database so that the autocorrelations between periods correspond to those observed historically. Just as in portfolio immunization, where the scenarios might have to satisfy a no arbitrage condition, in this case the strong correlation between the inflows in any pair of consecutive months must be respected.

An inflow scenario is also specified according to the degree of inflow level. For example, the scenario "*75% DRY*" could indicate that, *with a probability of 75%*, *the historically observed inflows are greater than or equal to the ones selected in the scenario.* Similarly, a scenario "*80% WET*" *indicates that the weekly inflows selected, in addition to satisfying the autocorrelation constraints, are greater than or equal to the historically observed ones 80% of the time and are within the same ranges as the observed values.*

Scenario Optimization requires one to assign probabilities to these scenarios. These may be purely subjective. For example, if the year has so far been very dry then we could assign a high probability to it remaining dry of the next month or so. We then experiment with various combinations of scenarios, such as, for next two months: "*DRY 80% with probability 0.9, DRY 60% with probability 0.1*"; and for all subsequent months: "*WET 50% with probability 0.5, DRY 75% with probability 0.5*", etc.

6. Conclusions

Scenario Optimization appears to be both cheap to implement and easy to understand. It represents a form of stochastic programming with, possibly nonlinear, simple recourse. We view Scenario Optimization as different from traditional approaches to stochastic optimization. It is a modelling framework and not a prescription for computing a solution. Much flexibility is still left to the modeller and thus the precise formulation of the tracking problem will usually differ from application to application.

Acknowledgements

We are indebted to Alan King for the numerous discussions we have had on this topic. His advice and feedback have been invaluable. This research was supported in part by a grant from the National Science and Engineering Research Council of Canada under grant number OGPIN007.

Notes

Patent protection for the Portfolio Immunization model described in this paper and other related models based on Scenario Optimization is now pending. For more information please contact:

Algorithmics Inc., 489 College Street, Toronto, Ontario, Canada M6G-1A5.

Appendix

STOCHASTIC DP VS. SCENARIO OPTIMIZATION

In comparing Stochastic Dynamic Programming (SDP) with Scenario Optimization (SOPT) we first need to state some facts about the nature of the hydroscheduling problem.

(1) *Stochasticity of inflows:*
- the distribution of reservoir inflows, conditioned on the current state of the system, may the approximated well for the immediate future (1 to 2 weeks);
- conditional distributions of future inflows are often very difficult, if not impossible to predict with any accuracy months in advance.

(2) *Individual nature of reservoirs:*
- operating characteristics of turbines differ greatly from reservoir to reservoir;
- geography of connected river systems may have an important effect; different parts of the system may have very different inflow characteristics;
- operating ranges and head/volume relationships may vary greatly among reservoirs.

SDP, because of the "curse of dimensionality", is forced to work with a single aggregated reservoir to represent an entire connected system. However, within this framework one may handle a reasonably fine discrete approximation of the possible scenarios.

SOPT, because of the number of scenario subproblems that one needs to solve may be extremely large, is forced to aggregate future inflow scenarios. However, because the single scenario subproblems are continuous, a high level of detail may be handled.

Stochastic Dynamic Programming (SDP) and Scenario Optimization (SOPT) may be viewed as two very different approaches to solving the same underlying stochastic model formulation.

In SDP all decision variables are discretized and an enumeration algorithm is used to compute an optimal solution at each time stage in a dynamic programming recursion. In SOPT, a series of deterministic optimization problems are solved, one for each scenario. In addition, a single coordinating problem is solved to find a single optimal trajectory.

For SDP, the problem dimension, and hence computation time, grows rapidly with the number of reservoirs and level of discretization. This is often referred to as the "curse of dimensionality".

For example, consider a case in which 100 discrete values are used to represent the level in a reservoir. If in addition, 100 discrete values are used to represent the possible inflow levels to the reservoir, then there are 10^4 possible combinations of reservoir levels and hydrological trends. If there are R reservoirs in the system then the number of possible combinations that have to be examined at each stage is 10^{4R}.

Thus for 1 reservoir there are 10 thousand states;
for 2 reservoirs there are 100 million states;
for 3 reservoirs there are 1 trillion states

that need to be examined at each stage in the dynamic programming recursion.

Clearly, since this quantity grows so rapidly with the number of reservoirs one has to aggregate the hydro network in order to obtain a model that is computationally feasible. In practice, this is usually done by creating an aggregate model consisting of one equivalent reservoir.

Thus there are two major sources of error in an SDP approach; error due to aggregation and hence loss of detail in the hydro system model and errors due to the discretization of state variables.

Scenario optimization on the other hand does not require discretization of the problem variables and uses the power of optimization techniques (linear and nonlinear programming) to handle the combinatorial structure of the problem efficiently.

The growth in problem size for SOPT is linear in the number of scenarios considered. For example, in Hidroeléctrica's Tajo system, with 10 reservoirs and 50 time stages a detailed optimization model (i.e. without aggregation), for a single scenario, would have approximately 1500 variables and 500 network flow constraints. If we were to consider 100 scenarios for future inflows into the system (an unrealistically large number), an SOPT model would require the solution of 100 (closely related) network optimization problems of the above size plus a single least squares problem to compute an optimal trajectory.

A more typical case would involve no more than 10 scenarios, in which case the SOPT approach would be computationally feasible.

Thus the error in an SOPT approach comes almost exclusively from the aggregation of future uncertain inflow scenarios.

In the hydro application, future inflow distributions are very difficult to predict beyond a horizon of 2 weeks. Since variations among the reservoir operating characteristics are so great, maintaining a high level of detail in the representation of reservoirs is essential whereas relatively little is lost in aggregating distant future inflow scenarios. Also, precipitation over catchment areas may vary considerably thereby making it important to maintain detail when representing the reservoir inflows.

For this reason SOPT appears to have an advantage over SDP. For the same or a lesser computational burden one may obtain a more detailed representation of the dynamics of the system while still accounting for the uncertainty of future inflows. In this case SOPT aggregates information that is poor while retaining detail where it is most needed. In contrast, to avoid the "curse of dimensionality" SDP aggregates reservoir detail while assuming that detail is available in the stochastic future inflows.

References

[1] F. Black, E. Derman and W. Toy, A one factor model of interest rates and its application to treasury bonds, Discussion Paper No. 1, Financial Strategies Group, Goldman Sachs and Co. (June 1988).

[2] R.S. Dembo, A. Chiarri, L. Paradinas and J. Gomez, Management of a hydrothermal energy system, Interfaces 20 (1990) 115-135.

[3] R.S. Dembo, Scenario immunization, Algorithmics Inc. Research Paper 89.02 (December 1989).
[4] R.S. Dembo and A.J. King, Tracking models for portfolio management, Algorithmics Inc. Research Paper 91.01 (February 1991).
[5] M.I. Kusy and W.I. Ziemba, A bank asset and liability management model, Oper. Res. 34 (1986) 356–376.
[6] J.L. Nazareth, Algorithms based upon generalized linear programming for stochastic programs with recourse, in: *Stochastic Programming Algorithms and Applications*, Lecture Notes on Control and Information Sciences, vol. 76 (Springer, 1986).
[7] R.T. Rockafellar and R. Wets, Scenarios and policy aggregation in optimization under uncertainty, IIASA Working Paper, WP-87-119 (December 1987); also, Math. Oper. Res., to appear.
[8] L.F. Silva, P.A. Arpe, L.P. Perry and M.V. Pereira, Coordinating the energy generation of the Brazilian system, Interfaces 16 (1986) 16-38.