Decomposition and Partitioning Methods for Multi-Stage Stochastic Linear Programs

John R. Birge

Technical Report No. 82-6

Department of Industrial and Operations Engineering
The University of Michigan
Ann Arbor, Michigan 48109

May, 1982

Decomposition and Partitioning Methods for Multi-Stage Stochastic Linear Programs

John R. Birge

Department of Industrial and Operations Engineering
The University of Michigan
Ann Arbor, Michigan 48109

Abstract

Multi-stage stochastic linear programs can be formulated for problems in financial planning, corporate decision making, economic policy analysis, and many other areas. Deterministic equivalent linear program representations of these problems, however, become excessively large. This paper presents methods for solving these problems based on decomposition and partitioning. Computational results on a set of practical test problems are reported to show the algorithms' potential usefulness.

Subject classification: 635 - Large scale systems programming
655 - Stochastic programming.

Many practical problems require that optimal decisions be made periodically over time. These problems can often be formulated as multi-stage (or dynamic) linear programs. The resulting programs have a staircase structure that may be exploited by decomposition (Ho and Manne [1974]), basis factorization (Fourer [1979]), partitioning (Rosen [1963]), and other computational techniques.

Difficulties may arise, however, in implementing solutions from these programs because of uncertainty about some parameters of the program. These random coefficients are often replaced by their expected values, but the solution of the corresponding expected value program may not be optimal for the stochastic program. In fact, no linear program which allows for only one value of each coefficient may lead to the optimal solution (Birge [1981]). In this case, a stochastic program must be solved to obtain the optimal solution. The deterministic equivalent linear program is usually very large, and standard solution procedures may prove very costly.

In this paper, we present two methods for solving these problems based on a nested decomposition (or outer linearization) approach and a piecewise linear partitioning strategy. Both methods can be viewed as approaches to the dynamic programming formulation presented in Section 1. The outer linearization approach of Section 2 is an extension of the Van Slyke and Wets [1969] method for the two stage problem. The piecewise strategy in Section 3 is presented as a partial resolution of a degeneracy problem inherent in the method of Section 2. Section 4 describes computational experience on a set of practical multi-stage problems taken from Ho and Loute [1981].

1. Problem Formulation

The stochastic linear program with recourse was first formulated by Beale [1955] and Dantzig [1955]. We consider the multi-stage version of this problem given by

$$\min \quad z = c_1x_1 + E_{\xi_2}[c_2x_2 + E_{\xi_3}[c_3x_3 + \ldots + E_{\xi_T}[c_Tx_T]\ldots]$$

subject to:

$$\begin{array}{lllll} A_1x_1 & & & & = b_1, \\ -B_1x_1 + & A_2x_2 & & & = \xi_2, \\ & -B_2x_2 & & & = \xi_3, \\ & & \ddots & & \vdots \\ & & & -B_{T-1}x_{T-1} + A_Tx_T & = \xi_T, \end{array}$$

$$x_t \geq 0, \; t = 1, 2, \ldots, T,$$

$$\xi_t \varepsilon \Xi_t, \; t = 1, 2, \ldots, T, \tag{1}$$

where c_t is a vector in R^{n_t} for $t = 1, 2, \ldots, T$, b_1 is a vector in R^{m_1}, ξ_t is a random m_2-vector defined on Ξ_t, $t = 2, \ldots, T$, and A_t and B_t are correspondingly dimensioned known real valued matrices. The vector x_1 represents decisions made in the first period, and vectors x_t represent decisions made in future periods dependent on the outcomes, $\xi_2, \ldots, \xi_t$, and previous decisions, $x_2, \ldots, x_t$. "E_{ξ_t}" represents the expectation with respect to ξ_t.

The decision process in a practical problem is to determine a first period decision which minimizes the sum of current current costs, c_1x_1, and the expected values of all future costs until some time T. Solution methods for this problem have been most thoroughly explored for the case of discrete distributions, where $\Xi_t = \{\xi_t^1, \xi_t^2,\ldots,\xi_t^{k_t}\}$, and T = 2. Dantzig and Madansky [1961] proposed that the dual of this problem be solved using Dantzig-Wolfe [1960] decomposition. Another approach in Kall [1979] and Strazicky [1980] is to solve the dual using a basis factorization scheme. Van Slyke and Wets [1969] also proposed applying outer linearization, or Benders' decomposition (Benders [1962]) to the primal problem. In Section 2 we present an extension of this method for the case of general T.

Other methods have been applied to the simple recourse problem in which T = 2 and $A_2 = [I \vdots - I]$. Everett and Ziemba [1979], for example, suggest a modification of the Frank-Wolfe algorithm, and Wets [1975] has suggested a method based on approximation by splines. Methods for this class of problems are important since the general form fits many problems in dynamic economic planning and financial planning.

For the multi-stage case, Beale, Forrest, and Taylor [1980] have considered a specific production model. In their model, several products are produced and either sold or kept in inventory according to the result of a stochastic demand vector realized after the production decision has been made. They show that an approximation procedure incorporated into a backward recursion can produce solutions very close to the exact dynamic program solutions.

We consider the general multi-stage problem (1) as a dynamic program with stages, 1, 2,..., T, and states $y_t = B_t x_t$, t = 1, 2,..., T-1. The value at t = T is defined as

$$z_T\ (y_{T-1}) = E_{\xi_T}[\zeta_T(y_{T-1},\ \xi_T)], \tag{2}$$

where

$$\begin{aligned} \zeta_T(y_{T-1},\ \xi_T) = \quad \min \quad & c_T\ x_T \\ \text{subject to: } & A_T\ x_T = y_{T-1} + \xi_T, \\ & x_T \geqq 0. \end{aligned} \tag{3}$$

The solution to (1) can then be found recursively by

$$z_t(y_{t-1}) = E_{\xi_t}[\zeta_t(y_{t-1},\ \xi_t)], \tag{4}$$

where

$$\begin{aligned} \zeta_t(y_{t-1}\ \xi_t) = \min_{(x_t, y_t)} \quad & c_t x_t + z_{t+1}(y_t) \\ \text{subj. to } & A_t x_t = \xi_t + y_{t-1}, \\ & B_t x_t = y_t, \\ & x_t \geqq 0. \end{aligned} \tag{5}$$

At t = 1, we define $y_0 = 0$ and $\xi_1 = b_1$, and obtain

$$\begin{aligned} z_1(0) = \min_{(x_1, y_1)} \quad & c_1 x_1 + z_2(y_1) \\ \text{subj. to } \quad & A_1 x_1 = b_1, \\ & B_1 x_1 = y_1, \\ & x_1 \geqq 0. \end{aligned} \tag{6}$$

The optimal solution, x_1^*, of (6) gives us the optimal set of first period decisions. Of course, the dimension of y_t may make direct implementation of this procedure impossible. In the following sections, we present outer linearization and piecewise linear approximation methods for this dynamic programming framework.

2. A Nested Benders' Decomposition Method.

Decomposition methods generally consist of <u>outer linearization</u>, as in Benders [1962], or <u>inner linearization</u>, as in Dantzig-Wolfe [1960] decomposition. Geoffrion [1970] has described these relationships and Kallio and Portius [1977] have discussed them in terms of dynamic linear programs. Ho and Manne [1974] have also formulated a <u>nested decomposition</u> approach of applying inner linearization to successive stages of dynamic linear programs.

In this section, we describe the application of outer linearization to the primal problem in (5). This procedure can also be described as applying inner linearization to the dual of (5). This method is based on the following equivalence:

<u>Proposition 1.</u> The mathematical program (5) is equivalent to:

$$\min \quad c_t x_t + Q_{t+1}(x_t) \tag{6}$$

$$\text{subj. to} \quad A_t x_t = \xi_t + B_{t-1} x_{t-1}, \tag{6.1}$$

$$D_t x_t \geqq \lambda_t, \tag{6.2}$$

$$x_t \geqq 0, \tag{6.3}$$

where $Q_{t+1}(x_t)$ is a real-valued convex function, D_t is an $m_{t+1} x n_t$ real-matrix, and λ_t is a real m_{t+1} vector.

Proof. See Wets [1966], Proposition 6. ■

The constraints (6.2) are known as feasibility cuts and are used to induce a feasible solution at stage t + 1. They can be constructed sequentially by noting that if the problem at t + 1 is not feasible for $\hat{x}_t$, a solution of (6), then for some $\hat{\xi}_{t+1}$ there does not exist $x_{t+1} \geq 0$ such that $A_{t+1} x_{t+1} = \hat{\xi}_{t+1} + B_t \hat{x}_t$, and $D_{t+1} x_{t+1} \geqq \hat{\lambda}_{t+1}$. In this case, there exists a vector $(\hat{\sigma}^1_{t+1}, \hat{\sigma}^2_{t+1})$ such that:

$$\hat{\sigma}^1_{t+1} A_{t+1} + \hat{\sigma}^2_{t+1} D_{t+1} \leq 0, \tag{7}$$

$$- \hat{\sigma}^2_{t+1} D_{t+1} \leq 0, \tag{8}$$

and $$\hat{\sigma}^1_{t+1} (\hat{\xi}_{t+1} + B_t \hat{x}_t) + \hat{\sigma}^2_{t+1} \hat{\lambda}_{t+1} > 0. \tag{9}$$

A constraint of the form,

$$- \hat{\sigma}^1_{t+1} B_t x_t \geq \hat{\sigma}^1_{t+1} \hat{\xi}_{t+1} + \hat{\sigma}^2_{t+1} \hat{\lambda}_{t+1}, \tag{10}$$

can then be added to (6.2) and D_t and λ_t can be formed row by row as consecutive solutions x^1_t, $x^2_t, \ldots$, are used to test feasibility at stage t + 1.

The convex objective function in (6) can also be handled by successive approximations. First, we rewrite (6) as:

$$\min \quad c_t x_t + \Theta_t \tag{11}$$

$$\text{subj. to} \quad A_t x_t = \xi_t + B_{t-1} x_{t-1}, \tag{11.1}$$

$$D_t x_t \geq \lambda_t \quad , \tag{11.2}$$

$$- Q(x_t) + \Theta_t \geq 0, \tag{11.3}$$

$$x_t \geqq 0. \tag{11.4}$$

We wish to show the following proposition.

Proposition 2. Constraint (11.3) in problem (11) for t = 1, 2,..., T, may be replaced by linear constraints of the form

$$-\bar{\pi}^{1,j}_{t+1} B_t x_t + \Theta_t \geq \bar{\rho}^{j}_{t+1}, \; j = 1, 2, \ldots, p, \qquad (12)$$

where $p \leq m_{t+1} < +\infty$, $\bar{\pi}^{1,j}_{t+1}$ is an $m_{t+1} + 1$ - vector, and $\bar{\rho}^{j}_{t+1}$ is a constant.

Proof. The significance of $\bar{\pi}^{1,j}_{t+1}$ and $\bar{\rho}_{t+1}$ will become clear in the proof. We assume that the proposition holds for all $t \geq \tau + 1$. It is vacuously true for $t = T$ since $Q_{T+1}(x_T) = 0$, and we have $\Theta_T \geq 0$.

Assume that x^1_τ is an optimal solution to (11) for $t = \tau$ in which $Q_{\tau+1}(x_\tau)$ has been replaced by a zero vector.

Let the optimal solutions of the stage $\tau + 1$ problems with $x_\tau = x^1_\tau$ be $(x^1_{\tau+1}(\xi_{\tau+1}), \Theta_{\tau+1}(\xi_{\tau+1}))$ and let the corresponding dual multipliers be $(\pi^{1,1}_{\tau+1}(\xi_{\tau+1}), \pi^{2,1}_{\tau+1}(\xi_{\tau+1}), \pi^{3,1}_{\tau+1}(\xi_{\tau+1}))$ where $\pi^{1,1}_{\tau+1}(\xi_{\tau+1} + B_\tau x^1_\tau) + \pi^{2,1}_{\tau+1}(\xi_{\tau+1})(\lambda_{\tau+1}) + \pi^{3,1}_{\tau+1}(\xi_{\tau+1})(\bar{\rho}_{\tau+2}(\xi_{\tau+1})) = c_{\tau+1} \; x^1_{\tau+1}(\xi_{\tau+1}) + \Theta^1_{\tau+1}(\xi_{\tau+1})$ and where $\bar{\rho}_{\tau+2}(\xi_{\tau+1})$ is the vector of $(\bar{\rho}^j_{\tau+2})$ for $j = 1, 2, \ldots, p$. If we have some other feasible solution to (11), $\hat{x}_\tau \neq x^1_\tau$, which leads to optimal multipliers, $(\hat{\pi}^1_{\tau+1}, (\xi_{\tau+1}), \hat{\pi}^2_{\tau+1}(\xi_{\tau+1}), \hat{\pi}^3_{\tau+1}(\xi_{\tau+1}))$, at $\tau + 1$, then

$$\hat{\pi}^1_{\tau+1}(\xi_{\tau+1})(\xi_{\tau+1} + B_\tau \hat{x}_\tau) + \hat{\pi}^2_{\tau+1}(\xi_{\tau+1})(\lambda_{\tau+1})$$

$$+ \; \hat{\pi}^3_{\tau+1}(\xi_{\tau+1})(\hat{\bar{\rho}}_{\tau+2} \; (\xi_{\tau+1}))$$

$$\geq \pi^{1,1}_{\tau+1}(\xi_{\tau+1})(\xi_{\tau+1} + B_\tau \; x^1_\tau) + \pi^{2,1}_{\tau+1}(\xi_{\tau+1})(\lambda_{\tau+1})$$

$$+ \; \hat{\pi}^3_{\tau+1}(\xi_{t+1})(\bar{\rho}_{\tau+2} \; (\xi_{\tau+1})) \qquad (13)$$

where $\hat{\bar{\rho}}_{\tau+2}$ consists of the union of linear constraints required at $\tau + 1$ for the x_τ^1 and $\hat{x}_\tau$ cases.

Since $Q_{\tau+1}(\hat{x}_\tau)$ is the expected value of the left hand side of (13), we obtain

$$Q_{\tau+1}(\hat{x}_\tau) \geq \bar{\rho}_{\tau+1}^{1,1} + \bar{\rho}_{\tau+1}^{2,1} + \bar{\rho}_{\tau+1}^{3,1} + \bar{\pi}_{\tau+1}^{1} B_\tau \hat{x}_\tau, \tag{14}$$

for all $\hat{x}_\tau$, where

$$\bar{\rho}_{\tau+1}^{1,1} = E_{\xi_{\tau+1}} [\pi_{\tau+1}^{1,1}(\xi_{\tau+1}) \cdot \xi_{\tau+1}], \tag{15.1}$$

$$\bar{\rho}_{\tau+1}^{2,1} = E_{\xi_{\tau+1}} [\pi_{\tau+1}^{2,1}(\xi_{\tau+1}) \cdot \lambda_{\tau+1}], \tag{15.2}$$

$$\bar{\rho}_{\tau+1}^{3,1} = E_{\xi_{\tau+1}} [\pi_{\tau+1}^{3,1}(\xi_{\tau+1}) \cdot \bar{\rho}_{\tau+2}(\xi_{\tau+1})], \tag{15.3}$$

and

$$\bar{\pi}_{\tau+1}^{1} = E_{\xi_{\tau+1}} [\pi_{\tau+1}^{1,1}(\xi_{\tau+1})]. \tag{16}$$

Letting $\bar{\rho}_{\tau+1}^{1} = \bar{\rho}_{\tau+1}^{1,1} + \bar{\rho}_{\tau+1}^{2,1} + \bar{\rho}_{\tau+1}^{3,1}$, we have that for all (x_τ, θ_τ) feasible in (11) for $t = \tau$,

$$\theta_\tau \geq \bar{\rho}_{\tau+1}^{1} + \bar{\pi}_{\tau+1}^{1} B_\tau x_\tau. \tag{17}$$

This inequality leads to a method for constructing the constraints in (12). We successively add a constraint of the form (12) whenever we solve (11) where linear constraints replace (11.3) and find an optimal solution $(x_\tau^j, \theta_\tau^j)$ such that $\theta_\tau^j < \bar{\rho}_{\tau+1}^{j} + \bar{\pi}_{\tau+1}^{j} B_\tau x_\tau$. The constraints we add lead to an outer linearization of $Q_{\tau+1}(x_\tau)$.

Murty [1968] showed that there is an optimal solution of (11) in which x_τ has at most $m_\tau + m_{\tau+1} + 1$ nonzero elements. Therefore, the greatest number of extra linear constraints in (11.3) required at the optimum is $m_{\tau+1} + 1$. This completes the induction step. ■

The modified version of (11) is then

$$
\begin{array}{lllll}
\min & c_t x_t + \theta_t & & & (18) \\
\text{subj. to} & A_t x_t & = \xi_t + B_{t-1}\, x_{t-1}, & & (18.1) \\
 & D_t x_t & \geq \lambda_t, & & (18.2) \\
 & -\bar{\pi}^{l,j}_{t+1} B_t x_t + \theta_t & \geq \bar{\rho}^{j}_{t+1}, & j = 1, 2, \ldots, p; & (18.3) \\
 & x_t & \geq 0. & & (18.4)
\end{array}
$$

The constraints in (18.3) are successively added to (18) until a solution, (x^*_t, θ^*_t) is found that satisfies $Q(x^*_t) \leq \theta^*_t$.

The outer linearization interpretation of the method can be seen in Figure 1. Linear supports are placed under the convex function $Q_{t+1}(x_t)$ until the value of θ^*_t from (18) is such that $\theta^*_t = Q_{t+1}(x^*_t)$. The method can also be shown to be an application of Dantzig-Wolfe decomposition to the dual of (5).

The complete algorithm consists of first obtaining a feasible solution and then successively solving (11) through the approximation (18) from period t back to period T.

Algorithm 1.

Step 0. Set up the following problem for all $t = 1, 2, \ldots, T$, and all ξ_t,

$$\min \quad c_t x_t + \Theta_t \tag{19}$$

$$\text{subject to } A_t x_t = \xi_t, \tag{19.1}$$

$$x_t \geq 0. \tag{19.2}$$

If there is no constraint on Θ_t, we let $\Theta_t^* = -\infty$ and let x_t^* solve the problem omitting Θ_t.

Step 1. Solve (19) for period 1 and $\xi_1 = b_1$. Let the resulting optimal solution be x_1^*. If the period 1 problem is infeasible, stop, the problem is infeasible. If not, replace each ξ_2 at period (2) by $\xi_2 + B_1 x_1^*$. Solve each of these period 2 problems. If any is infeasible, then add a feasibility cut as in (10) to the period 1 problem in (19) and repeat. If every period 2 problem is feasible , then let $t = 2$ and go to Step 2.

Step 2. For all distinct solutions, $\{x_t^*(\xi_t)\}$ at period t, solve the t + 1 period problem by replacing each ξ_{t+1} in (19) for t + 1 by $\xi_{t+1} + x_t^*(\xi_t)$. If any of these is infeasible, then add a constraint of type (10) to the corresponding period t problem and solve that problem. If the corresponding period t problem is infeasible, then let t = t - 1 and (a) repeat Step 2 if t - 1 > 1, or (b) go to Step 1 if t - 1 = 1. If the period t problem is feasible, repeat Step 2.

If $t \leq T - 2$ and all t + 1 period problems are feasible then let t = t + 1 and repeat Step 2. If t = T - 1 and all t + 1 period problems are feasible, then set $\hat{\Theta}_t = -\infty$ to correspond with every solution, $\hat{x}_t$, t = 1, 2,..., T-1, and go to Step 3.

Step 3. Find $\bar{\pi}_{t+1}^{1}(x_t^*)$ and $\bar{\rho}_{t+1}(x_t^*)$ for the current solutions of the t + 1 period problems for every period t solution x_t^*. If $\Theta_t^* < \bar{\rho}_{t+1}(x_t^*) + \bar{\pi}_t^{1}(x_t^*) \cdot B_t x_t^*$, then add a constraint of type (12') to the period t problem (19) for which x_t^* was optimal. Do this for each x_t^* and solve each of these period t problems. Again, use the resulting solutions and solve all period t + 1 problems. If t < T - 1, let t = t + 1 and go to Step 2. If t = T - 1, repeat Step 3.

If $\Theta_t^* = \bar{\rho}_{t+1}(x_t^*) + \bar{\pi}_t^{1}(x_t^*) \cdot B_t x_t^*$ for all x_t^*, then if t > 1, let t = t - 1 and repeat Step 3, and, if t = 1, then, Stop, the problem is optimal.

Steps 1 and 2 in the algorithm are called the foreward pass through all periods to obtain feasibility. Step 3 is the backward pass in a dynamic programming type recursion. We note that we have made no mention of unbounded solutions in the algorithm. This may be resolved by the procedure in Van Slyke and Wets [1969], but, in practice, we replace (19.2) by $\ell_t \leq x_t \leq u_t$ where $u_t(i) < + \infty$ for all i and $\ell_t(i) > - \infty$ for all i. This bounded variable problem then does not admit an unbounded solution, although some adjustment for nonbasic variables at their upper bounds must be made in the feasibility cuts (10).

We note also that the number of possible solutions in one period may be infinite if ξ_t is allowed a continuous distribution. In practice, we assume that $\Xi = \{\xi_t^1, \xi_t^2, \ldots, \xi_t^{k_t}\}$ and that $P(\xi_t = \xi_t^i) = p_t^i$. We also assume that ξ_t and $\xi_{t'}$ are independent random variables for all $t \neq t'$. With these assumptions, it is clear that the algorithm proceeds back from t to t - 1 after a finite number of steps and that hence, the algorithm converges finitely.

The repeated structure of the constraint matrix, A_t, in (19) for all realizations of the random variable, ξ_t, allows us to avoid solving each problem independently. Instead, a single problem may be solved to yield a dual feasible basis and other problems for which this basis is also primal feasible may be found parametrically. This procedure is described in Garstka and Rutenberg [1973]. In our implementation of Algorithm 1, this has only been done for t = T since the problems for t < T may have additional constraints. They could also be handled, however, using a more complex data structure, in which, a working basis corresponding to A_t was stored and updated for different additional constraints.

Another difficulty that arises in implementations of Algorithm 1 is that many passes between a master problem at period t and a subproblem at period t + 1 may occur without creating substantial changes in the period t problem solution. One reason for this extra work is that additional constraints on the period t problem allow more than m_t period t columns to become basic. This causes degeneracy in the period t + 1 problem. Dantzig and Abrahamson [1980] observed this behavior in their experiments on deterministic multi-stage linear programs. They noticed that the period t problem basis changed very slightly from one pass to the next, and they theorized that this process could be improved with more interaction between the period t master problem and the period t + 1 subproblems. The basic degeneracy results are in the following propositions.

Proposition 3. If a constraint of the form (10) is added to (19) and is binding for some optimal solution x_t^*, then every primal basic feasible solution of (19) for t + 1, where (19.1) is replaced by

$$A_{t+1}\, x_{t+1} = \hat{\xi}_{t+1} + B_t x_t^*, \tag{20}$$

is degenerate.

Proof: For the binding feasibility cut, we have

$$-\hat{\sigma}_t^1\, B_t\, x_t^* = \hat{\sigma}_t^1\, \hat{\xi}_{t+1}, \tag{21}$$

where it is assumed that $\hat{\lambda}_{t+1} = 0$. Now, let $A_{t+1}^{\hat{B}}$ be a feasible basis of (19) with right hand side $\hat{\xi}_{t+1} + B_t\, x_t^*$ and let $x_{t+1}^{\hat{B}} = (A_{t+1}^{\hat{B}})^{-1}(\hat{\xi}_{t+1} + B_t x_t^*)$. Multiply each side of (20) by $\hat{\sigma}_t^1$ to obtain

$$\hat{\sigma}_t^1\, A_{t+1}^{\hat{B}}\, x_{t+1}^{\hat{B}} = \hat{\sigma}_t^1\, (\, \hat{\xi}_{t+1} + B_t x_t^*) = 0. \tag{22}$$

Since $\hat{\sigma}^1_t$ was found previously such that $\hat{\sigma}^1_t A_{t+1} \leq 0$, then $\hat{\sigma}^1_t A^{\hat{B}}_{t+1} \leq 0$. $\hat{\sigma}^1_t A^{\hat{B}}_{t+1} \neq 0$, however, since $A^{\hat{B}}_{t+1}$ is a basis and $\hat{\sigma}^1_t \neq 0$. Hence, there exists some $x^{\hat{B}}_{t+1}(j) = 0$. ■

Proposition 4. If two constraints of the form in (12) are added to (19) and are binding at an optimal solution, (x^*_t, θ^*_t), then every set of solutions of (19) for t + 1 with right hand side in (19.1), $\xi_{t+1} + B_t x^*_t$, such that $\theta^*_t = \bar{\rho}_{t+1} + \bar{\pi}^1_{t+1} B_t x^*_t$, includes a degenerate solution for some $\xi_{t+1} \in \Xi_{t+1}$.

Proof. Let the binding constraints be

$$- \bar{\pi}^{1,1}_{t+1} B_t x^*_t + \theta^*_t = \bar{\rho}^1_{t+1}, \tag{23}$$

and

$$- \bar{\pi}^{1,2}_{t+1} B_t x^*_t + \theta^*_t = \bar{\rho}^2_{t+1}. \tag{24}$$

Let the optimal bases at t + 1 be the set $\{A^B_{t+1}(\xi_{t+1})\}$ and let the corresponding prices be $\{\pi^{1,*}_{t+1}(\xi_{t+1})\}$ (where again additional constraints have been omitted in the period t + 1 problem). This set of prices must be distinct from one of the sets which generated $\bar{\pi}^{1,1}_{t+1}$ or $\bar{\pi}^{1,2}_{t+1}$ because, when constraint (24) was added to (19) for period t, (23) had to be satisfied and (24) had to be violated. Without loss of generality, let the current set of prices be distinct from those in (23) and the same as those which generated (24).

Let $c^B_{t+1}(\xi_{t+1}) = \pi^{1,*}_{t+1}(\xi_{t+1})\ A^B_{t+1}(\xi_{t+1})$ and let $x^B_{t+1}(\xi_{t+1}) = (A^B_{t+1}(\xi_{t+1}))^{-1}\ (\xi_{t+1} + B_t\ x^*_t)$. It follows that

$$\begin{aligned}
&E_{\xi_{t+1}}\ [\pi^{1,*}_{t+1}(\xi_{t+1})(A^B_{t+1}(\xi_{t+1})\ x^B_{t+1}(\xi_{t+1}))] \\
&= E_{\xi_{t+1}}\ [\pi^{1,*}_{t+1}(\xi_{t+1})(\xi_{t+1} + B_t\ x^*_t)] \\
&= \Theta^*_t \\
&= E_{\xi_{t+1}}\ [\pi^{1,1}_{t+1}(\xi_{t+1})(\xi_{t+1} + B_t x^*_t)] \\
&= E_{\xi_{t+1}}\ [\pi^{1,1}_{t+1}(\xi_{t+1})(A^B_{t+1}(\xi_{t+1})\ x^B_{t+1}(\xi_{t+1}))]. \qquad (25)
\end{aligned}$$

From (25) and the definition of $c^B_{t+1}(\xi_{t+1})$, we obtain

$$\int_{\Xi_{t+1}} (c^B_{t+1}(\xi_{t+1}) - \pi^{1,1}_{t+1}(\xi_{t+1})\ A^B_{t+1}(\xi_{t+1}))\ x^B_{t+1}(\xi_{t+1})\ dF(\xi_t) = 0, \quad (26)$$

where F is the distribution function of ξ_{t+1}.

Now for $\pi^{1,1}_{t+1}(\xi_{t+1})$ dual feasible, $c^B_{t+1}(\xi_{t+1}) - \pi^{1,1}_{t+1}(\xi_{t+1})\ A^B_{t+1}(\xi_{t+1}) \geq 0$ and $x^B_{t+1}(\xi_{t+1}) \geq 0$. Since $\{\pi^{1,1}_{t+1}(\xi_{t+1})\} \neq \{\pi^{1,*}_{t+1}(\xi_{t+1})\}$ as assumed above, there exists some $\hat{\xi}_{t+1}$ such that $c^B_{t+1}(\hat{\xi}_{t+1}) \neq \pi^{1,1}_{t+1}(\hat{\xi}_{t+1})\ A^B_{t+1}(\hat{\xi}_{t+1})$ and, therefore, $x^B_{t+1}(\hat{\xi}_{t+1})$ must be degenerate. ■

This degeneracy can force the period t + 1 subproblem to be overly restricted. Abrahamson [1980] suggests an alternative in the deterministic model in which columns are passed foreward from the master problem to the subproblem. This method does not directly apply to the stochastic case as it is not known beforehand which value of ξ_{t+1} will obtain a different optimal basis. A proposal to allow for this coupling is presented in Birge [1980] but it creates an additional intermediary problem between the master problem of period t and the period t + 1 subproblems. This difficulty motivated the development of the following algorithm based on the piecewise linearity of $Q_{t+1}(x_t)$ for discrete random vectors ξ_{t+1}.

3. A Piecewise Linear Partitioning Method.

The approach followed in this section is restricted to the case where all random variables are discrete. We again will optimize between a master problem in period t and a subproblem in period t + 1, but the two problems are more closely connected. The basic idea is similar to that used by Rosen [1963]. A convex function is minimized on a subregion within the feasible region and a boundary or interior point solution is found. If the optimum occurs at a boundary point then optimizations are performed in all adjacent subregions to that point. If no adjacent region yields a better value or if the solution is interior, then the procedure stops and the current point is optimal.

We begin the development of this procedure by discussing the basic optimality conditions between the t + 1 subproblem and the period t master problem. Let the set of optimal bases of (6) for t + 1 be

$\{A^{B,1}_{t+1}, A^{B,2}_{t+1}, \ldots, A^{B,k_{t+1}}_{t+1}\}$. We note that:

$$Q_{t+1}(x^*_t) = \sum_{i=1}^{k_{t+1}} p^i_{t+1}(c^B_{t+1})(A^{B,i}_{t+1})^{-1}(\xi^i_{t+1} + B_t x^*_t), \tag{27}$$

for x^*_t an optimal solution from period t. Because of the dual feasibility of $c_{t+1}(A^{B,i}_{t+1})^{-1} = \pi^i_{t+1}$,

$$Q_{t+1}(x_t) = \sum_{i=1}^{k_{t+1}} p^i_{t+1}(c^B_{t+1})(A^{B,i}_{t+1})^{-1}(\xi^i_{t+1} + B_t x_t), \tag{28}$$

as long as

$$(A^{B,i}_{t+1})^{-1} (\xi^i_{t+1} + B_t x_t) \geq 0. \tag{29}$$

Hence, $Q_{t+1}(x_t)$ is linear while (29) is satisfied. We can, therefore, solve (6) on linear pieces of $Q_{t+1}(x_t)$ by solving:

$$\min \quad z^0_t = [c_t + \sum_{j=1}^{k_{t+1}} p^j_{t+1} \, c^B_{t+1}(A^{B,j}_{t+1})^{-1} B_t] \, x_t + \sum_{j=1}^{k_{t+1}} p^j_{t+1} c^B_{t+1}(A^{B,j}_{t+1})^{-1} \, \xi^j_{t+1} \tag{30}$$

subject to

$$A_t x_t = \xi_t + B_{t-1} \, x^*_{t-1}, \tag{30.1}$$

$$(A^{B,j}_{t+1})^{-1} \, B_t x_t \geq - (A^{B,j}_{t+1})^{-1} \, \xi^j_{t+1}, \; j = 1,\ldots,k_{t+1}, \tag{30.2}$$

$$x_t \geq 0. \tag{30.3}$$

The solution of (30) allows the choice of x_t to be optimal in one step for a given set of bases in t + 1. The advantage over the decomposition method is that this process may avoid switching among period t solutions which correspond to a single set of period t + 1 bases.

After solving (30), we obtain an optimal solution, x_t^0, and consider the set of binding constraints in (30.2). We define

$$\mathcal{T} = \{(i, j): [(A_{t+1}^{B,j})^{-1} B_t](i, \cdot) \cdot x_t^0 = - [(A_{t+1}^{B,j})^{-1}\xi_{t+1}^{j}](i))\} \quad (31)$$

where i corresponds to the binding row and j corresponds to the binding subproblem at t + 1, a row of a matrix M is denoted M(i, •), and a column is denoted M(•, i). These are the places of degeneracy in the subproblems. We next order the pairs (i, j) ε $\mathcal{T}$ lexicographically and write $\mathcal{T} = \{\tau_1, \tau_2, \ldots, \tau_p\}$.

If $\mathcal{T} = \phi$ then $x_t^* = x_t^0$ and $c_t x_t^0 + Q_{t+1}(x_t^0) \leq c_t x_t + Q_{t+1}(x_t)$ for all x_t by convexity, so x_t^* is optimal in (6).

If $\mathcal{T} \neq \phi$, consider $\tau_1 = (i_1, j_1)$. We have

$$x_{t+1}^{B,j_1} = (A_{t+1}^{B,j_1})^{-1} (\xi_{t+1}^{j_1} + B_t x_t^0), \quad (32)$$

and

$$x_{t+1}^{B,j_1}(i_1) + [A_{t+1}^{B,j_1})^{-1}(i_1 \cdot) \cdot A_{t+1}^{N,j_1}]\, x_{t+1}^{N,j_1} = 0, \quad (33)$$

where x_{t+1}^{N,j_1} are nonbasic variables in subproblem j_1 at t + 1 and A_{t+1}^{N,j_1} is the partition of A_{t+1} corresponding to those variables.

We want to force $x_{t+1}^{B,j_1}(i_1)$ from the basis in order to enter an adjacent feasible region. We find the entering variables $x_{t+1}(s)$, according to the dual simplex criterion (Dantzig [1963]),

$$\frac{c_{t+1}(s) - \pi^{j_1}_{t+1} A_{t+1}(i_1,s)}{- \bar{A}_{t+1}(i_1,s)} = \min_{\substack{j \ni \bar{A}(i_1,j) < 0 \\ t+1}} \frac{c_{t+1}(j) - \pi^{j_1}_{t+1}(i_1, j)}{- \bar{A}_{t+1}(i_1,j)}$$

where $\bar{A}_{t+1}(i_1,j) = [(A^{B,j_1})^{-1} A_{t+1}(\cdot, j)](i_1)$. (34)

Pivoting $x_{t+1}(s)$ in to replace $x^{B,j_1}_{t+1}(i_1)$ results in a new basis $A^{\bar{B},j_1}_{t} = A^{B,j_1}_{t+1} \cdot \eta$, where η is an elementary matrix. We next form the following auxiliary problem by replacing A^{B,j_1}_{t+1} with $A^{\bar{B},j_1}_{t+1}$:

$$\min \quad \bar{z}^0_t = [c_t + \sum_{j=1}^{k_{t+1}} p^j_{t+1} c^{\bar{B}}_{t+1} (A^{\bar{B},j}_{t+1})^{-1} B_t] x_t$$
$$+ \sum_{j=1}^{k_{t+1}} p^j_{t+1} c^{\bar{B}}_{t+1} (A^{\bar{B},j}_{t+1})^{-1} \xi^j_{t+1} \quad (35)$$

subject to

$$A_t x_t = \xi_t + B_{t-1} x^*_{t-1}, \quad (35.1)$$

$$(A^{\bar{B},j}_{t+1})^{-1} B_t x_t \geq - (A^{\bar{B},j}_{t+1})^{-1} \xi^j_{t+1}, \; j = 1,\ldots,k_{t+1}, \quad (35.2)$$

$$x_t \geq 0. \quad (35.3)$$

where $A^{\bar{B},j}_{t+1} = A^{B,j}_{t+1}$ for all $j \neq j_1$.

Problem (35) has a different objective function and some different constraints from (30). x^0_t is feasible in (35), but may not be optimal. The algorithm proceeds by solving (35) to obtain $\bar{x}^0_t$. If $\bar{z}^0_t < z^0_t$, then we let $x^1_t = \bar{x}^0_t$ and $z^1_t = \bar{z}^0_t$ form a new set $\mathcal{T}$ for the binding constraints in (35.2) and solve a new auxiliary problem for $\bar{z}^1_t$ compared with z^1_t.

If $\bar{z}_t^0 \geq z_t^0$, then we proceed to $(i_2, j_2) \in \mathcal{T}$ and form a new auxiliary problem pivoting to an adjacent basis in j_2. We increment the index on z_t whenever a better solution is found until we obtain some z_t^M such that no $\bar{z}_t^M < z_t^M$. In this case, the solution x_t^M is optimal in (6).

The procedure creates a decreasing sequence of values, $z_t^0 > z_t^1 > \cdots > z_t^M$, so that, because there are only a finite number of basis sets for the period t + 1 problems, it must converge finitely. An example of the route of the optimization is in Figure 2. The problem is solved along piecewise linear sections of the curve $Q_{t+1}(x_t)$. It terminates when no improving direction is found.

Substantial reductions in storage can also be obtained in this procedure by only using distinct bases in the sets of constraints, (30.2) and (35.2). Many period t + 1 problems may have the same optimal basis. We let $\mathcal{B} = \{A_{t+1}^{B,\ell},\ \ell = 1, 2, \ldots, q\}$, be the set of distinct optimal bases in period t + 1. We define

$$P_{t+1}(\ell) = \sum_{j \in J(\ell)} p_{t+1}^j \tag{36}$$

where $J(\ell) = \{j:\ A_{t+1}^{B,j} = A_{t+1}^{B,\ell}\}$.

We also need

$$\rho_{t+1}^{\ell}(i) = -[(A_{t+1}^{B,\ell})^{-1}(\xi_{t+1}^{\hat{j}})](i) = \max_{j \in J(\ell)} \{-[(A_{t+1}^{B,\ell})^{-1}\xi_{t+1}^{j}](i)\}, \tag{37}$$

the ith component of the right hand side in (30.2), and we store $\hat{j}\ (i, \ell)$ for all i and ℓ. (30) then becomes:

$$\min \quad z_t^0 = [c_t + \sum_{\ell=1}^{q} P^{\ell}_{t+1} c^{B}_{t+1} (A^{B,\ell}_{t+1})^{-1} B_t]\, x_t + \sum_{j=1}^{k_{t+1}} p^{j}_{t+1} c^{B}_{t+1} (A^{B,j}_{t+1})^{-1} \xi^{j}_{t+1} \qquad (38)$$

subject to

$$A_t x_t = \xi_t + B_{t-1} x^*_{t-1}\,, \qquad (38.1)$$

$$(A^{B,\ell}_{t+1})^{-1} B_t x_t \geq \rho^{\ell}_{t+1};\ \ell = 1,\ldots,\ q; \qquad (38.2)$$

$$x_t \geqq 0. \qquad (38.3)$$

The elements in $\mathcal{T}$ are sorted by $(i, \hat{j}(i, \ell))$ for binding constraints on (38.2) and, whenever a new basis is found, it must be added to $\mathcal{B}$. We note that the constant term in the objective of (38) must be updated for each individual ξ^{j}_{t+1} but that this does not affect the optimization. The same iterative procedure as above can again be applied. The modification can, however, substantially reduce the number of constraints. A parametric procedure can again be used to efficiently generate the set of optimal bases in the period t + 1 problem.

The entire multi-stage piecewise algorithm proceeds as in the decomposition by first obtaining a feasible solution and then iterating backward to obtain optimality from period t to T for t = T - 1, T - 2,...,1.

<u>Algorithm 2</u>.

<u>Step 1</u>. Follow Steps 0, 1, and 2 of Algorithm 1 to obtain a feasible solution for the entire problem. Let t = T - 1, let $j_t = 1$ for all t and go to Step 2.

Step 2. Solve the j_t problem of the form in (38) for $\xi_t = \xi_t^{j_t}$ and $x^*_{t-1} = x_{t-1}(j_2,\dots,j_{t-1})$ in (38.1), where $x_{t-1}(j_2,\dots,j_{t-1})$ is the current solution corresponding to realizations, $\xi_2^{j_2},\dots,\ \xi_{t-1}^{j_t}$, of the random variables. Construct $\mathcal{T}$, let $\ell = 1$, and go to Step 3.

Step 3. If $\ell > p$, (the cardinality of $\mathcal{T}$),

$\Lambda = \max\{1;\ \lambda \text{ s.t. } j_\lambda < k_\lambda \text{ for } 2 \leq \lambda \leq t - 1\}$.

If $\Lambda = 1$ and $t = 1$, stop, the current problem is optimal. If $\Lambda = 1$, and $t > 1$ let $t = t - 1$, $j_t = 1$ for all t and go to Step 2. If $\Lambda > 1$, let $j_\lambda = j_\lambda + 1$, let $j_t = 1$ for all $t > \lambda$ and go to Step 2.

If $\ell < p$, set up the auxiliary problem corresponding to (38) with $A_{t+1}^{\bar{B},\hat{j}}$ substituted for $A_t^{\bar{B},j}$. If the resulting objective value $\bar{z}_t^0 < z_t^0$, let $z_t^0 = \bar{z}_t^0$, $x_t^0 = \bar{x}_t^0$, and go to Step 2. If $\bar{z}_t^0 = z_t^0$, let $\ell = \ell + 1$ and repeat Step 3.

Unboundedness is taken care of as in Algorithm 2 by bounding variables. The algorithm converges to an optimal or infeasible solution because each period t master-period t + 1 subproblem is solved finitely, and t is decreased whenever all period t problems have been solved.

We note that if several bases are adjacent to a single basis in forming the auxiliary problem, then we consider each of these and correspondingly increase p to allow for duplication. We note also that primal feasibility is maintained throughout Steps 2 and 3 of Algorithm 2 so that additional feasibility constraints are not required after Step 1. Furthermore, we observe that the bases for period t problems steadily increase in size as they are passed back from period T because they must ensure feasibility for all subsequent periods. In this case, then, Algotithm 2 solves much larger problems than Algorithm 1 as additional periods are incorporated into the solution. This also makes it difficult to use the method of only keeping distinct bases for any period but the last.

Algorithm 2, therefore, has the advantage of maintaining some connections between master and subproblems and of preserving primal feasibility. It may, however, be less efficient on these problems because of their increased size. In the next section, we compare these algorithms on a set of test problems.

4. Computational Results.

Two FORTRAN programs, NDST3 and PCST2, were written to implement Algorithms 1 and 2, respectively. The linear programs in (19) and (38) were solved using LPM-1 as a subroutine. LPM-1 was written by J. A. Tomlin and revised by G. Kochman at the Systems Optimization Laboratory, Stanford University. It employs compact storage of non-zero entries in the coefficient matrix, performs an LU-decomposition of the basis, and uses a merit counting sort to maintain sparsity in the inverse(Pfefferkorn,and Tomlin [1976]).

NDST3 currently handles problems with up to $T = 4$ periods and up to 1,000 realizations of the right hand side vector. Each period t problem is limited to 350 rows and 600 columns. The number of nonzero elements in any of these problems is less than or equal to 3,000. NDST3 solves period T problems by parametrizing the right hand side and only storing distinct bases. For $t < T$, however, each problem is solvedindependently.

PCST2 currently applies only to the 2 stage case. In attempting to implement the code for $T > 2$, it was found that storage requirements were greater for the same size problem if the problem was broken into $T > 2$ periods, instead of solving it with $T = 2$. The size of the individual problems at period t would also not be reduced. In the case of Algorithm 2, then, it was assumed that unless a data structure is used which enables one to keep only distinct bases for $t < T$, the problem should only be solved in the two-stage format. PCST2 allows for 125 realizations of the right hand side subproblems and handles the same size subproblems as does NDST3. PCST2 also uses parameterization in the second period and stores only distinct bases.

The test problems used are taken from Ho and Loute [1981]. They are practical deterministic multi-stage problems with staircase structure, as in the constraint matrix of (1). These problems were chosen instead of randomly generated problems because it was thought they would be more representative of actual applications. For comparative purposes, all problems were limited to 3 periods and the number of right hand side realizations was limited to 8. This size limitation enabled the problem to be solved using a standard linear programming code on the full deterministic equivalent problem and provided a check on solution values. It shouldbe noted, however, that the greatest advantage from the techniques would probably come from problems whose full deterministic equivalent linear programs were too large for standard mathematical programming packages.

The problems were taken from the following areas of application:

Problem	Application
SC205	Economic development,
SCAGR7	Agricultural planning,
SCRS8	Dynamic energy modeling,
SCSD8	Structural design optimization,
SCTAP1	Dynamic traffic assignment,
SCFXM1	Production scheduling.

Their statistics, where "Totals" refers to the size of the deterministic equivalent linear program, appear in Table I.

The values in Table I are the result of solutions using the IBM mathematical programming package, MPS/360, (IBM [1973]), on The University of Michigan's Amdahl 470/V8 computer. The number of iterations for MPS/360 to obtain the optimal solution appears in Table II. Unfortunately, CPU second times for the solutions are not available for MPS/360 on the Michigan MTS operating system. The number of iterations are given only for rough comparisons with the results for NDST3 and PCST2. It should be noted that these iterations are on larger problems than those in NDST3 and PCST2.

The test problems were each solved by NDST3 and PCST2 on the Amdahl 470/V8 using the standard FORTRAN-G compiler without optimization (The University of Michigan Computing Center [1980]). The results for breaking the problem into 2 and 3 period problems for NDST3 are both reported in TableII. CPU seconds do not include problem input or solution output. "Iterations" are the number of simplex pivots and "passes" are the number of distinct subproblems solved. (The number of passes is augmented only once for t = T.) Results for PCST2 are also listed in Table II.

In the first five problems, NDST3 with T = 2 performed the best. There appeared to be no advantage from furthering decomposing the problem into T = 3 periods or from employing the partitioning approaching in PCST2. These results should not be considered conclusive, however, considering the sample size and the absence of advanced storage techniques for the t < T cases. The basic reason for good performance relative to MPS/360 appears to be the elimination of repeated solutions for different subproblems. In the first five problems, degeneracy also did not appear to be a problem in the nested decomposition approaches. PCST2 reduced the number of passes but not enough to compensate for its solving largerproblems.

The example SCFXM1 displayed some behavior that can be especially confounding for both NDST3 and PCST2. In the process of obtaining a feasible solution, a large number of feasibility cuts (10) were generated These cuts were very dense and hence required the storage of a large number of nonzero elements. In fact, after 23 cuts had been placed on the period 1 problem, the maximum number of nonzero elements, 3,000, had been exceeded. Since the period 1 problem could conceivably have m_2 completely dense cuts of this type, an upper bound on the number of elements for this period 1 problem is $m_2 \cdot n_1$ or 8,100 in this case. This number is greater than the number of nonzeroes in the deterministic equivalent linear program.

This situation is one of the potential difficulties involved in using decomposition or partitioning procedures. It does not appear to be common as smaller problems have not exhibited this property (Birge [1980]), but, in solving programs as in (1), some check on the growth of nonzeroes should be incorporated into the solution procedure. It should also be noted that, in general, the subproblems solved in NDST3 are much smaller than the full linear program. For T = 2, if

(i) $\nu(1)$ = number of nonzeroes in A_1,

(ii) $\nu(2)$ = number of nonzeroes in B_1, and

(iii) $\nu(3)$ = number of nonzeroes in A_3,

then the linear program has $\nu(1) + k_2(\nu(2) + \nu(3))$ nonzero elements(plus the objective row and right hand sides), whereas the period 1 problem in NDST3 has at most $\nu(1) + n_1 \cdot \Pi$ nonzeroes, where Π is the number of passes. When Π is small as in the first five examples, this value is small.

5. Summary

Two algorithms for multi-stage stochastic linear programs were presented. The methods were based on decomposition and partitioning procedures that are common in large-scale mathematical programming. Computer codes for these algorithms were applied to a set of practical problems from applications in a variety of areas. The methods solved five of these problems in substantially fewer iterations than a standard linear programming approach, but one problem required an excessive number of nonzero elements to be added to the master problem. The results seem to indicate that gains in efficiency come from reducing the number of bases stored for the subproblems and from decreasing the problem sizes.

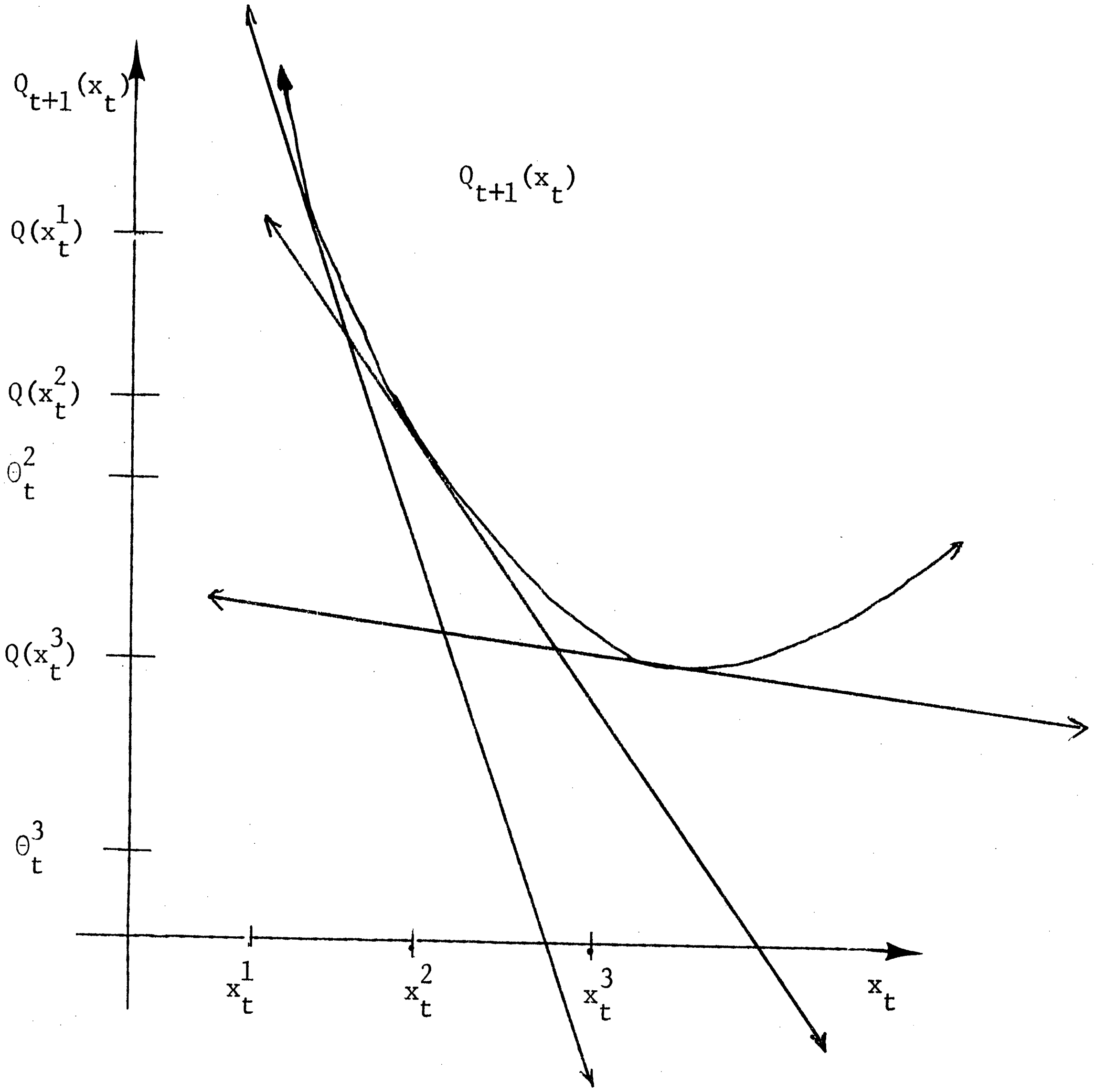

Figure 1. Outer Linearization of $Q_{t+1}(x_t)$.

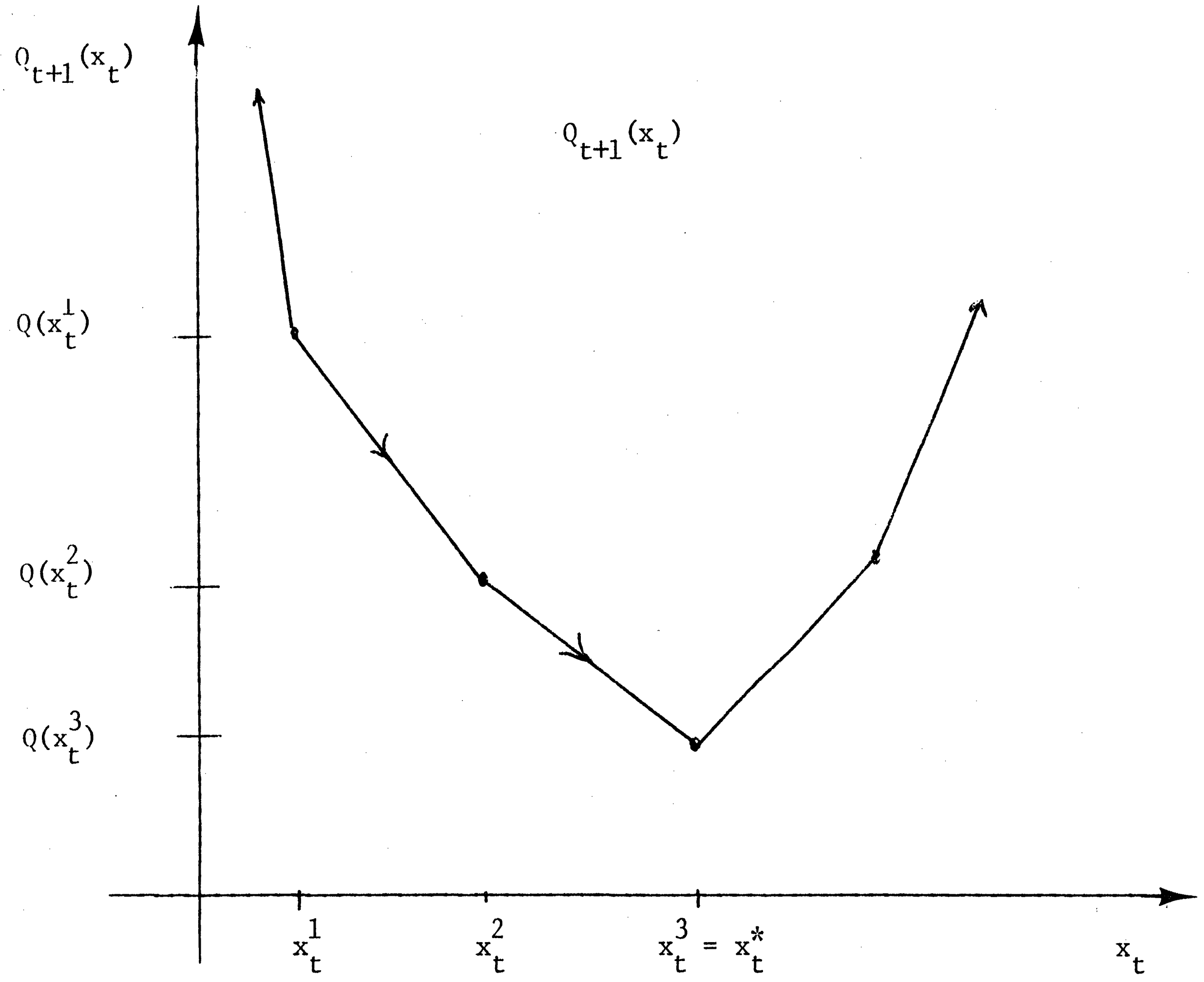

Figure 2. Piecewise Linear Paths of $Q_{t+1}(x_t)$.

Table I

Problem Statistics

	Within period					Totals			Optimal
	Rows		Cols						
Problem	Max	Min	Max	Min	Rows	Cols	Els[1]	Den[2]	Value
SC205	13	11	14	11	190	380	701	.97	60.4
SCAGR7	15	19	20	20	320	660	1693	.80	832903.5
SCRS8	31	28	38	37	254	342	1133	1.3	123.4
SCTAP1	30	30	48	48	511	817	3814	.91	240.5
SCSD8	10	10	70	70	171	1191	4867	2.4	16.5
SCFXM1	92	82	126	99	1277	1915	7952	.33	2877.6

1 - Nonzero elements

2 - % of elements that are nonzero.

Table II

Results

	MPS/ 360	NDST3(T = 2)			NDST3(T = 3)			PCST2		
	It[a]	CPUs[b]	It	Ps[c]	CPUs	It	Ps	CPUs	It	Ps
SC205	139	.07	24	8	.26	28	25	.13	24	7
SCAGR7	409	.49	100	12	.61	117	49	.93	69	11
SCRS8	203	.11	28	4	.29	23	19	.25	28	3
SCSD8	381	.15	11	4	.39	52	19	.34	11	3
SCTAP1	337	.43	64	4	.76	92	35	.71	63	4
SCFXM1	1742	d	d	d	e	e	e	d	d	d

a - Number of simplex method pivots.

b - CPU seconds excluding problem input and solution output.

c - Number of subproblems solved.

d - Exceeded nonzero element limit after 40 passes and over 800 iterations. Last solution was 25% from optimal.

e - Exceeded nonzero element limit after 49 passes and over 1,000 iterations. Last solution was 35% from optimal.

REFERENCES

Abrahamson, P.G. 1980. A Nested Decomposition Approach for Solving Staircase Structured Linear Programs. In Proceedings of the Workshop on Large-Scale Linear Programming. IIASA, Laxenburg, Austria, June 2-6.

Beale, E.M.L. 1955. On Minimizing an Convex Function Subject to Linear Inequalities. Journal of the Royal Stat. Soc. B17, 173-184.

Beale, E.M.L., J.J.H. Forrest and C.J. Taylor. 1980. Multi-time Period Stochastic Programming. In Stochastic Programming, M.A.H. Dempter (ed.). Academic Press, New York.

Benders, J.F. 1962. Partitioning Procedures for Solving Mixed-variables Programming Problems. Numerische Mathematik 4, 238-252.

Birge, J.R. 1980. Solution Methods for Stochastic Dynamic Linear Programs. Technical Report SOL 80-29, Systems Optimization Laboratory, Stanford University.

Birge, J.R. 1981. The Value of the Stochastic Solution in Stochastic Linear Programs with Fixed Recourse. Technical Report 81-10, Department of Industrial and Operations Engineering, The University of Michigan. Also to appear in Mathematical Programming.

Dantzig, G.B. 1955. Linear Programming under Uncertainty. Mgmt. Sci. 1, 197-206.

Dantzig, G.B. 1963. Linear Programming and Extensions. Princeton University Press, Princeton, NJ

Dantzig, G.B. and A. Madansky . 1961. On the Solution of Two-stage Linear Programs under Uncertainty. In Proceedings, 4th Berkeley Symp. on Math. Stat. and Prob. U.C. Press, Berkeley.

Dantzig, G.B. and P. Wolfe. 1960. Decomposition Principle for Linear Programs. Opns. Res. 8, 101-111.

Everett, R. and W.T. Ziemba. 1979. Two-Period Stochastic Programs with Simple Recourse. Opns. Res. 27, 485-502.

Fourer, R. 1979. Solving Staircase L.P.'s by the Simplex Method. Technical Report SOL 70-18, Systems Optimization Laboratory, Stanford University.

Garstka, S. and D. Ruthenberg. 1973. Computation in Discrete Stochastic Programming with Recourse. Opns. Res. 21, 112-122.

Geoffrion, A.M. 1970. Elements of Large-Scale Mathematical Programming. Mgmt. Sci. 16, 652-675.

Ho, J.K. and E. Loute. 1981. A Set of Staircase Linear Programming Test Problems. Math. Program. 20, 245-250.

Ho, J.K. and A.S. Manne. 1974. Nested Decomposition for Dynamic Models. Math. Program. 6, 121-140.

IBM. 1973. Mathematical Programming System/360 Application Description (H20-0136-3).

Kall, P. 1979. Computational Methods for Solving Two-Stage Stochastic Linear Programming Problems. Journal of App. Math. and Physics 30, 261-271.

Kallio, M. and E.L. Porteus. 1977. Decomposition of Arborescent Linear Programs. Math. Program. 13, 348-355.

Murty, K.G. 1968. Two-Stage Linear Programming under Uncertainty: a Basic Property of the Optimal Solution. Z.F. Wahrs. u. Ver. Geb. 10, 284-288.

Pfefferkon, C.E. and J.A. Tomlin. 1976. Design of a Linear Programming System for ILLIAC IV. Technical Report SOL 78-7, Systems Optimization Laboratory, Stanford University.

Rosen, J.B. 1973. Convex Partition Programming. In Recent Advances in Mathematical Programming, R.L. Graves and P. Wolfe (eds.). McGraw-Hill, New York.

Strazicky, B. 1980. Some Results Concerning an Algorithm for the Discrete Recourse Problem. In Stochastic Programming, M.A.H. Dempster (ed.). Academic Press, New York.

The University of Michigan Computing Center. 1980, MTS, Volumne 6, FORTRAN in MTS.

Van Slyke, R. and R. Wets. 1969. L-Shaped Linear Programs with Applications to Optimal Control and Stochastic Linear Programs. SIAM J. on App. Math. 17, 638-663.

Wets, R. 1966. Programming under Uncertainty : the Solution Set. SIAM J. on App. Math. 14, 1143-1151.

Wets, R. 1975. Solving Stochastic Programs with Simple Recourse, II. Working Paper, Department of Mathematics, University of Kentucky.