Benders' Decomposition Methods for Structured Optimization, including Stochastic Optimization

Robert M. Freund

April 29, 2004

1 Block Ladder Structure

We consider the solution of a linear optimization model of the basic format:

$$\begin{array}{rcccccl}
\text{minimize}_{x,y} & c^T x & + & f^T y & & \\
\text{s.t.} & Ax & & & = & b \\
 & Bx & + & Dy & = & d \\
 & x \geq 0 & & y \geq 0 \ . & &
\end{array}$$

Here the variables x can be thought of as stage-1 decisions, governed by constraints $Ax = b, x \geq 0$ and with cost function $c^T x$. Once the x variables have been chosen, the y variables are chosen next, subject to the constraints $Dy = d - Bx, y \geq 0$ and with cost function $f^T y$.

We also consider the more complex format associated with two-stage optimization under uncertainty:

$$\begin{array}{rccccccccccl}
\underset{x, y_1, \ldots, y_K}{\text{minimize}} & c^T x & + & \alpha_1 f_1^T y_1 & + & \alpha_2 f_2^T y_2 & + & \cdots & + & \alpha_K f_K^T y_K & & \\
\text{s.t.} & Ax & & & & & & & & & = & b \\
 & B_1 x & + & D_1 y_1 & & & & & & & = & d_1 \\
 & B_2 x & & & + & D_2 y_2 & & & & & = & d_2 \\
 & \vdots & & & & & & \ddots & & & & \vdots \\
 & B_K x & & & & & & & + & D_K y_K & = & d_K \\
 & \multicolumn{5}{l}{x, y_1, y_2, \ldots, y_K \geq 0 \ .} & & & & & &
\end{array}$$

Actually, each of these formats can be thought of as a special case of the other format. The more complex format above is known as “block-ladder”, and is illustrated in Figure 1.

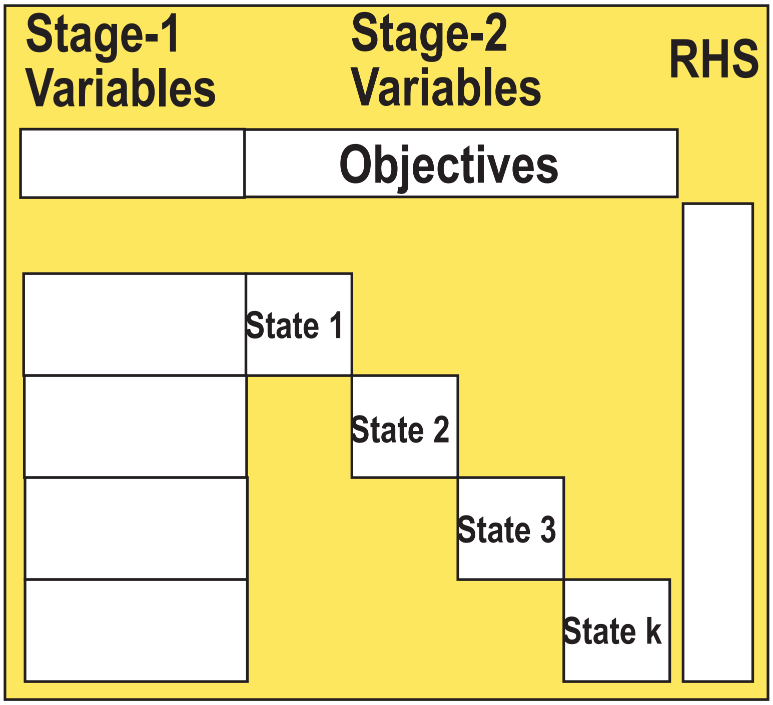

Figure 1: Block ladder structure of two-stage stochastic linear optimization.

2 Reformulation of the Basic Model

The basic model format is:

$$\begin{array}{rrrcrcl}
\text{VAL} & = \text{minimum}_{x,y} & c^T x & + & f^T y & & \\
 & \text{s.t.} & Ax & & & = & b \\
 & & Bx & + & Dy & = & d \\
 & & x \geq 0 & & y \geq 0 \; , & &
\end{array}$$

whose optimal objective value is denoted by VAL. We re-write this as:

$$\begin{array}{rlrcl} \text{VAL} & = \text{minimum}_x & c^T x + z(x) & & \\ & \text{s.t.} & Ax & = & b \\ & & x \geq 0 \ , & & \end{array}$$

where:

$$\begin{array}{rlrcl} \text{P2}: \ z(x) & = \text{minimum}_y & f^T y & & \\ & \text{s.t.} & Dy & = & d - Bx \\ & & y \geq 0 \ . & & \end{array}$$

We call this problem "P2" because it represents the stage-2 decision problem, once the stage-1 variables x have been chosen. Applying duality, we can also construct $z(x)$ through the dual of P2, which we denote by D2:

$$\begin{array}{rlrcl} \text{D2}: \ z(x) & = \text{maximum}_p & p^T(d - Bx) & & \\ & \text{s.t.} & D^T p & \leq & f \ . \end{array}$$

The feasible region of D2 is the set:

$$\mathcal{D}_2 := \left\{ p \mid D^T p \leq f \right\} \ ,$$

whose extreme points and extreme rays can be enumerated:

$$p^1, \ldots, p^I$$

are the extreme points of $\mathcal{P}_2$, and

$$r^1, \dots, r^J$$

are the extreme rays of $\mathcal{P}_2$.

If we use an algorithm to solve D2, exactly one of two cases will arise: D2 is unbounded from above, or D2 has an optimal solution. In the first case, the algorithm will find that D2 is unbounded from above, and it will return one of the extreme rays $\bar{r} = r^j$ for some j, with the property that

$$(r^j)^T(d - Bx) > 0$$

in which case

$$z(x) = +\infty \ .$$

In the second case, the algorithm will solve D2, and it will return one of the extreme points $\bar{p} = p^i$ for some i as well as the optimal objective function value $z(x)$ which will satisfy:

$$z(x) = (p^i)^T(d - Bx) = \max_{k=1,\dots,I} (p^k)^T(d - Bx) \ .$$

Therefore we can re-write D2 again as the following problem:

$$\begin{array}{llllll} \text{D2}: & z(x) & = \text{minimum}_z & & z & \\ & & \text{s.t.} & (p^i)^T(d - Bx) & \leq & z & i = 1, \dots, I \\ & & & (r^j)^T(d - Bx) & \leq & 0 & j = 1, \dots, J \ . \end{array}$$

This problem simply states that the value of D2 is just the dual objective function evaluated at the best extreme point of the dual feasible region, provided that the objective function makes an obtuse angle with every extreme ray of the dual feasible region.

We now take this reformulation of $z(x)$ and place it in the orginal problem:

$$\begin{array}{llllll} \text{FMP}: & \text{VAL} & = \text{minimum}_{x,z} & c^T x + z & & \\ & & \text{s.t.} & Ax & = & b \\ & & & x \geq 0 & & \\ & & & (p^i)^T(d - Bx) & \leq & z & i = 1, \ldots, I \\ & & & (r^j)^T(d - Bx) & \leq & 0 & j = 1, \ldots, J \ . \end{array}$$

We call this problem FMP for the *f*ull *m*aster *p*roblem. Comparing FMP to the original version of the problem, we see that we have eliminated the variables y from the problem, we have added a single scalar variable z, and we have also added a generically extremely large number of constraints.

3 Delayed Constraint Generation

The idea behind delayed constraint generation is to try to solve FMP using only a small subset of the constraints, and to check afterward whether any of the non-included constraints are violated.

Consider the *r*estricted *m*aster *p*roblem RMP composed of only k of the extreme point / extreme ray constraints from FMP:

$$\begin{array}{llllll} \text{RMP}^k: & \text{VAL}^k & = \text{minimum}_{x,z} & c^T x + z & & \\ & & \text{s.t.} & Ax & = & b \\ & & & x \geq 0 & & \\ & & & (p^i)^T(d - Bx) & \leq & z & i = 1, \ldots, k - l \\ & & & (r^j)^T(d - Bx) & \leq & 0 & j = 1, \ldots, l \ . \end{array}$$

We solve this problem, obtaining the optimal objective value VAL^k and the solution $\bar{x}, \bar{z}$ that is optimal for this problem. We first observe that VAL^k is a lower bound on VAL, i.e.:

$$\mathrm{VAL}^k \quad \leq \quad \mathrm{VAL} \ .$$

In order to check whether the solution $\bar{x}, \bar{z}$ is optimal for the full master problem, we must check whether $\bar{x}, \bar{z}$ violates any of the non-included constraints. We check for this by solving the linear optimization problem:

$$\begin{array}{lllllll} \mathrm{Q}(\bar{x}): & = & \mathrm{maximum}_p & p^T(d - B\bar{x}) & = & \mathrm{minimum}_y & f^T y \\ & & \mathrm{s.t.} & D^T p \leq f & & \mathrm{s.t.} & Dy = d - B\bar{x} \\ & & & & & & y \geq 0 \ . \end{array}$$

If $\mathrm{Q}(\bar{x})$ is unbounded, the algorithm will return an extreme ray $\bar{r} = \ r^j$ for some j, where $\bar{x}$ will satisfy:

$$(r^j)^T(d - B\bar{x}) > 0 \ .$$

We therefore have found that $\bar{x}$ has violated the constraint:

$$(r^j)^T(d - Bx) \leq 0 \ ,$$

and so we add this constraint to RMP and re-solve RMP.

If $\mathrm{Q}(\bar{x})$ has an optimal solution, the algorithm will return an optimal extreme point $\bar{p} = \ p^i$ for some i, as well as the optimal solution $\bar{y}$ to the minimization problem of $\mathrm{Q}(\bar{x})$.

Now notice first from Q($\bar{x}$) that $(\bar{x}, \bar{y})$ is feasible for the original problem. Therefore, if UB is a previously computed upper bound on VAL, we can update the upper bound and our "best" feasible solution as follows:

$$\text{If } \; c^T\bar{x} + f^T\bar{y} \; \leq \; \text{UB}, \; \text{ then } \; \text{BESTSOL} \; \leftarrow (\bar{x}, \bar{y}) \; .$$

$$\text{UB} \leftarrow \min\{\text{UB}, c^T\bar{x} + f^T\bar{y}\} \; .$$

Also, notice that if:

$$(p^i)^T(d - B\bar{x}) > \bar{z} \; ,$$

then we have found that $\bar{x}, \bar{z}$ has violated the constraint:

$$(p^i)^T(d - Bx) \leq z \; ,$$

and so we add this constraint to RMP and re-solve RMP.

If $(p^i)^T(d - B\bar{x}) \leq \bar{z}$, then we have:

$$\max_{k=1,\ldots,I}(p^k)^T(d - B\bar{x}) = (p^i)^T(d - B\bar{x}) \leq \bar{z} \; ,$$

and so $\bar{x}, \bar{z}$ is feasible and hence optimal for FMP, and so $(\bar{x}, \bar{y})$ is optimal for the original problem, and so we terminate the algorithm.

By our construction of upper and lower bounds on VAL, we can also terminate the algorithm whenever:

$$\text{UB} - \text{VAL}^k \leq \varepsilon$$

for a pre-specified tolerance ε.

3.1 Delayed Constraint Generation Algorithm

Here is a formal description of the algorithm just described.

0. Set LB$= -\infty$ and UB$= +\infty$.

1. A typical iteration starts with the relaxed master problem RMP^k, in which only k constraints of the full master problem FMP are included. An optimal solution $\bar{x}, \bar{z}$ to the relaxed master problem is computed. We update the lower bound on VAL:

$$\text{LB} \leftarrow \text{VAL}^k \ .$$

2. Solve the subproblem Q$(\bar{x})$:

$$\begin{array}{lrlcrr} \text{Q}(\bar{x}): & \text{maximum}_p & p^T(d - B\bar{x}) & = & \text{minimum}_y & f^T y \\ & \text{s.t.} & D^T p \le f & & \text{s.t.} & Dy = d - B\bar{x} \\ & & & & & y \ge 0 \ . \end{array}$$

3. If Q$(\bar{x})$ has an optimal solution, let $\bar{p}$ and $\bar{y}$ be the primal and dual solutions of Q$(\bar{x})$. If $\bar{p}^T(d - B\bar{x}) \le \bar{z}$, then $\bar{x}, \bar{z}$ satisfies all constraints of FMP, and so $\bar{x}, \bar{z}$ is optimal for FMP. Furthermore, this implies that $(\bar{x}, \bar{y})$ is optimal for the original problem.

If $\bar{p}^T(d - B\bar{x}) > \bar{z}$, then we add the constraint:

$$\bar{p}^T(d - Bx) \le z$$

to the restricted master problem RMP.

Update the upper bound and the best solution:

$$\text{If } \; c^T\bar{x} + f^T\bar{y} \; \leq \; \text{UB}, \;\; \text{then } \; \text{BESTSOL } \; \leftarrow (\bar{x}, \bar{y})$$

$$\text{UB} \leftarrow \min\{\text{UB}, c^T\bar{x} + f^T\bar{y}\}$$

If UB $-$ LB $\leq \varepsilon$, then terminate. Otherwise, return to step **1.**

4. If $\mathrm{Q}(\bar{x})$ is unbounded from above, let $\bar{r}$ be the extreme ray generated by the algorithm that solves $\mathrm{Q}(\bar{x})$. Add the constraint:

$$\bar{r}^T(d - Bx) \leq 0$$

to the restricted master problem RMP, and return to step **1.**

This algorithm is known formally as Benders' decomposition. The Benders' decomposition method was developed in 1962 [2], and is described in many sources on large-scale optimization and stochastic programming. A general treatment of this method can be found in [3, 4].

The algorithm can be initialized by first computing any $\bar{x}$ that is feasible for the first stage, that is, $A\bar{x} = b, \bar{x} \geq 0$, and by choosing the value of $\bar{z}$ to be $-\infty$ (or any suitably small number). This will force $(\bar{x}, \bar{z})$ to violate some constraints of FMP.

Notice that the linear optimization problems solved in Benders' decomposition are "small" relative to the size of the original problem. We need to solve RMP^k once every outer iteration and we need to solve $\mathrm{Q}(\bar{x})$ once every outer iteration. Previous optimal solutions of RMP^k and $\mathrm{Q}(\bar{x})$ can serve as a starting basis for the current versions of RMP^k and $\mathrm{Q}(\bar{x})$ if we use the simplex method. Because interior-point methods do not have the capability of starting from a previous optimal solution, we should think of Benders' decomposition as a method to use in conjunction with the simplex method primarily.

Incidentally, one can show that Benders' decomposition method is the same as Dantzig-Wolfe decomposition applied to the dual problem.

4 Benders' Decomposition for Problems with Block-Ladder Structure

Suppose that our problem has the complex block-ladder structure of the following optimization model that we see for two-stage optimization under uncertainty:

$$
\begin{array}{rlcccccccccl}
\mathrm{VAL} & = \underset{x, y_1, \ldots, y_K}{\text{minimize}} & c^T x & + & \alpha_1 f_1^T y_1 & + & \alpha_2 f_2^T y_2 & + & \cdots & + & \alpha_K f_K^T y_K & \\
 & \text{s.t.} & Ax & & & & & & & & & = \ b \\
 & & B_1 x & + & D_1 y_1 & & & & & & & = \ d_1 \\
 & & B_2 x & & & + & D_2 y_2 & & & & & = \ d_2 \\
 & & \vdots & & & & & & \ddots & & & \ \ \vdots \\
 & & B_K x & & & & & & & + & D_K y_K & = \ d_K \\
 & & \multicolumn{9}{l}{x, y_1, y_2, \ldots, y_K \geq 0 \ .} &
\end{array}
$$

whose optimal objective value is denoted by VAL. We re-write this as:

$$
\begin{array}{rlccl}
\mathrm{VAL} & = \text{minimum}_x & c^T x + \sum_{\omega=1}^{K} \alpha_\omega z_\omega(x) & & \\
 & \text{s.t.} & Ax & = & b \\
 & & x \geq 0 \ , & &
\end{array}
$$

where:

$$\begin{array}{llll} \mathrm{P2}_\omega: \quad z_\omega(x) & = \mathrm{minimum}_{y_\omega} & f_\omega^T y_\omega & \\ & \text{s.t.} & D_\omega y_\omega & = d_\omega - B_\omega x \\ & & y_\omega \geq 0 \ . & \end{array}$$

We call this problem "P2$_\omega$" because it represents the stage-2 decision problem under scenario ω, once the stage-1 variables x have been chosen. Applying duality, we can also construct $z_\omega(x)$ through the dual of P2$_\omega$, which we denote by D2$_\omega$:

$$\begin{array}{llll} \mathrm{D2}_\omega: \quad z_\omega(x) & = \mathrm{maximum}_{p_\omega} & p_\omega^T(d_\omega - B_\omega x) & \\ & \text{s.t.} & D_\omega^T p_\omega & \leq f_\omega \ . \end{array}$$

The feasible region of D2$_\omega$ is the set:

$$\mathcal{D}_2^\omega := \left\{ p_\omega \mid D_\omega^T p_\omega \leq f_\omega \right\} \ ,$$

whose extreme points and extreme rays can be enumerated:

$$p_\omega{}^1, \ldots, p_\omega^{I_\omega}$$

are the extreme points of $\mathcal{D}_2^\omega$, and

$$r_\omega{}^1, \ldots, r_\omega^{J_\omega}$$

are the extreme rays of $\mathcal{D}_2^\omega$.

If we use an algorithm to solve D2$_\omega$, exactly one of two cases will arise: D2$_\omega$ is unbounded from above, or D2$_\omega$ has an optimal solution. In the first case, the algorithm will find that D2$_\omega$ is unbounded from above, and it will return one of the extreme rays $\bar{r}_\omega = r_\omega^j$ for some j, with the property that

$$(r_\omega^j)^T(d_\omega - B_\omega x) > 0$$

in which case

$$z_\omega(x) = +\infty \ .$$

In the second case, the algorithm will solve D2$_\omega$, and it will return one of the extreme points $\bar{p}_\omega = p_\omega^i$ for some i as well as the optimal objective function value $z_\omega(x)$ which will satisfy:

$$z_\omega(x) = (p_\omega^i)^T(d_\omega - B_\omega x) = \max_{k=1,\ldots,I_\omega} (p_\omega^k)^T(d_\omega - B_\omega x) \ .$$

Therefore we can re-write D2$_\omega$ again as the following problem:

$$\begin{array}{llllll} \mathrm{D2}_\omega : & z_\omega(x) & = \text{minimum}_{z_\omega} & & z_\omega & \\ & & \text{s.t.} & (p_\omega^i)^T(d_\omega - B_\omega x) & \leq \quad z_\omega & i = 1, \ldots, I_\omega \\ & & & (r_\omega^j)^T(d_\omega - B_\omega x) & \leq \quad 0 & j = 1, \ldots, J_\omega \ . \end{array}$$

This problem simply states that the value of D2$_\omega$ is just the dual objective function evaluated at the best extreme point of the dual feasible region, provided that the objective function makes an obtuse angle with every extreme ray of the dual feasible region.

We now take this reformulation of $z_\omega(x)$ and place it in the orginal problem:

$$
\begin{array}{rrcll}
\text{FMP}: \ \text{VAL} = \text{minimum} & c^T x + \sum_{\omega=1}^{K} \alpha_\omega z_\omega & & & \\
x, z_1, \ldots, z_K & & & & \\
\text{s.t.} & Ax & = & b & \\
 & x \geq 0 & & & \\
 & (p_\omega^i)^T (d_\omega - B_\omega x) & \leq & z_\omega & i = 1, \ldots, I_\omega, \ \omega = 1, \ldots, K \\
 & (r_\omega^j)^T (d_\omega - B_\omega x) & \leq & 0 & j = 1, \ldots, J_\omega, \ \omega = 1, \ldots, K \ .
\end{array}
$$

We call this problem FMP for the *f*ull *m*aster *p*roblem. Comparing FMP to the original version of the problem, we see that we have eliminated the vector variables $y_1, \ldots, y_K$ from the problem, we have added K scalar variables z_ω for $\omega = 1, \ldots, K$, and we have also added a generically extremely large number of constraints.

As in the basic model, we consider the *r*estricted *m*aster *p*roblem RMP composed of only k of the extreme point / extreme ray constraints from FMP:

$$
\begin{array}{rrcll}
\text{RMP}^k: \ \text{VAL}^k = \text{minimum} & c^T x + \sum_{\omega=1}^{K} \alpha_\omega z_\omega & & & \\
x, z_1, \ldots, z_K & & & & \\
\text{s.t.} & Ax & = & b & \\
 & x \geq 0 & & & \\
 & (p_\omega^i)^T (d_\omega - B_\omega x) & \leq & z_\omega & \text{for some } i \text{ and } \omega \\
 & (r_\omega^j)^T (d_\omega - B_\omega x) & \leq & 0 & \text{for some } i \text{ and } \omega \ ,
\end{array}
$$

where there are a total of k of the inequality constraints. We solve this problem, obtaining the optimal objective value VAL^k and the solution $\bar{x}, \bar{z}_1, \ldots, \bar{z}_K$

that is optimal for this problem. We first observe that VAL^k is a lower bound on VAL, i.e.:

$$\mathrm{VAL}^k \quad \leq \quad \mathrm{VAL}\ .$$

In order to check whether the solution $\bar{x}, \bar{z}_1, \ldots, \bar{z}_K$ is optimal for the full master problem, we must check whether $\bar{x}, \bar{z}_1, \ldots, \bar{z}_K$ violates any of the non-included constraints. We check for this by solving the K linear optimization problems:

$$\begin{array}{rllll} \mathrm{Q}_\omega(\bar{x}) := \mathrm{maximum}_{p_\omega} & p_\omega^T(d_\omega - B_\omega \bar{x}) & = & \mathrm{minimum}_{y_\omega} & f_\omega^T y_\omega \\ \text{s.t.} & D_\omega^T p_\omega \leq f_\omega & & \text{s.t.} & D_\omega y_\omega = d_\omega - B_\omega \bar{x} \\ & & & & y_\omega \geq 0\ . \end{array}$$

If $\mathrm{Q}_\omega(\bar{x})$ is unbounded, the algorithm will return an extreme ray $\bar{r}_\omega = r_\omega^j$ for some j, where $\bar{x}$ will satisfy:

$$(r_\omega^j)^T(d_\omega - B_\omega \bar{x}) > 0\ .$$

We therefore have found that $\bar{x}$ has violated the constraint:

$$(r_\omega^j)^T(d_\omega - B_\omega x) \leq 0\ ,$$

and so we add this constraint to RMP.

If $\mathrm{Q}_\omega(\bar{x})$ has an optimal solution, the algorithm will return an optimal extreme point $\bar{p}_\omega = p_\omega^i$ for some i, as well as the optimal solution $\bar{y}_\omega$ of the minimization problem of $\mathrm{Q}_\omega(\bar{x})$.

If

$$(p_\omega^i)^T(d_\omega - B_\omega \bar{x}) > \bar{z}_\omega\ ,$$

then we have found that $\bar{x}, \bar{z}_\omega$ has violated the constraint:

$$(p^i_\omega)^T(d_\omega - B_\omega x) \leq z_\omega \ ,$$

and so we add this constraint to RMP.

If $\mathrm{Q}_\omega(\bar{x})$ has finite optimal objective function values for all $\omega = 1, \ldots, K$, then $\bar{x}, \bar{y}_1, \ldots, \bar{y}_K$ satisfies all of the constraints of the original problem. Therefore, if UB is a previously computed upper bound on VAL, we can update the upper bound and our "best" feasible solution as follows:

$$\text{If} \quad c^T\bar{x} + \sum_{\omega=1}^{K} \alpha_\omega {f_\omega}^T \bar{y}_\omega \ \leq \ \text{UB}, \quad \text{then} \quad \text{BESTSOL} \quad \leftarrow (\bar{x}, \bar{y}_1, \ldots, \bar{y}_K) \ .$$

$$\text{UB} \leftarrow \min\{\text{UB}, c^T\bar{x} + \sum_{\omega=1}^{K} \alpha_\omega {f_\omega}^T \bar{y}_\omega\} \ .$$

If $(p^i_\omega)^T(d_\omega - B_\omega \bar{x}) \leq \bar{z}_\omega$ for all $\omega = 1, \ldots, K$, then we have:

$$\max_{l=1,\ldots,I_\omega} (p^l_\omega)^T(d_\omega - B_\omega \bar{x}) = (p^i_\omega)^T(d_\omega - B_\omega \bar{x}) \leq \bar{z}_\omega \quad \text{for all } \omega = 1, \ldots, K \ ,$$

and so $\bar{x}, \bar{z}_1, \ldots, \bar{z}_K$ satisfies all of the constraints in FMP. Therefore, $\bar{x}, \bar{z}_1, \ldots, \bar{z}_K$ is feasible and hence optimal for FMP, and so $(\bar{x}, \bar{y}_1, \ldots, \bar{y}_K)$ is optimal for the original problem, and we can terminate the algorithm.

Also, by our construction of upper and lower bounds on VAL, we can also terminate the algorithm whenever:

$$\text{UB} - \text{VAL}^k \leq \varepsilon$$

for a pre-specified tolerance ε.

4.1 The Delayed Constraint Generation Algorithm for Problems with Block-Ladder Structure

Here is a formal description of the algorithm just described for problems with block-ladder structure.

0. Set LB$= -\infty$ and UB$= +\infty$.

1. A typical iteration starts with the relaxed master problem RMP^k, in which only k constraints of the full master problem FMP are included. An optimal solution $\bar{x}, \bar{z}_1, \ldots, \bar{z}_K$ to the relaxed master problem is computed. We update the lower bound on VAL:

$$\mathrm{LB} \leftarrow \mathrm{VAL}^k \ .$$

2. For $\omega = 1, \ldots, K$, solve the subproblem $\mathrm{Q}_\omega(\bar{x})$:

$$\begin{array}{lllll} \mathrm{Q}_\omega(\bar{x}) := \mathrm{maximum}_{p_\omega} & p_\omega^T(d_\omega - B_\omega \bar{x}) & = & \mathrm{minimum}_{y_\omega} & f_\omega^T y_\omega \\ \text{s.t.} & D_\omega^T p_\omega \leq f_\omega & & \text{s.t.} & D_\omega y_\omega = d_\omega - B_\omega \bar{x} \\ & & & & y_\omega \geq 0 \ . \end{array}$$

- If $\mathrm{Q}_\omega(\bar{x})$ is unbounded from above, let $\bar{r}_\omega$ be the extreme ray generated by the algorithm that solves $\mathrm{Q}(\bar{x})$. Add the constraint:

$$\bar{r}_\omega^T(d_\omega - B_\omega x) \leq 0$$

  to the restricted master problem RMP.
- If $\mathrm{Q}_\omega(\bar{x})$ has an optimal solution, let $\bar{p}_\omega$ and $\bar{y}_\omega$ be the primal and dual solutions of $\mathrm{Q}_\omega(\bar{x})$. If $(p_\omega^i)^T(d_\omega - B_\omega \bar{x}) > \bar{z}_\omega$, then we add the constraint:

$$\bar{p}_\omega^T(d_\omega - B_\omega x) \leq z_\omega$$

to the restricted master problem RMP.

3. If $(p^i_\omega)^T(d_\omega - B_\omega \bar{x}) \leq \bar{z}_\omega$ for all $\omega = 1, \ldots, K$, then $(\bar{x}, \bar{y}_1, \ldots, \bar{y}_K)$ is optimal for the original problem, and the algorithm terminates.

4. If $\mathrm{Q}_\omega(\bar{x})$ has an optimal solution for all $\omega = 1, \ldots, K$, then update the upper bound on VAL as follows:

$$\text{If} \;\; c^T\bar{x} + \sum_{\omega=1}^{K} \alpha_\omega {f_\omega}^T \bar{y}_\omega \;\leq\; \text{UB}, \;\; \text{then} \;\; \text{BESTSOL} \;\; \leftarrow (\bar{x}, \bar{y}_1, \ldots, \bar{y}_K)\;.$$

$$\text{UB} \quad \leftarrow \quad \min\{\text{UB}, c^T\bar{x} + \sum_{\omega=1}^{K} \alpha_\omega {f_\omega}^T \bar{y}_\omega\}\;.$$

If UB $-$ LB $\leq \varepsilon$, then terminate. Otherwise, add all of the new constraints to RMP and return to step **1.**

In this version of Benders' decomposition, we might add as many as K new constraints per major iteration.

5 The Power Plant Investment Model

Recall the power plant investment model described earlier. The full model is:

$$
\begin{aligned}
\min_{x,y} \quad & \sum_{i=1}^{4} c_i x_i + \sum_{\omega=1}^{125} \alpha_\omega \sum_{i=1}^{5}\sum_{j=1}^{5}\sum_{k=1}^{15} 0.01 f_i(\omega) h_j y_{ijk\omega} \\
\text{s.t.} \quad & \sum_{i=1}^{4} c_i x_i \le 10,000 \qquad \text{(Budget constraint)} \\
& x_4 \le 5.0 \qquad \text{(Hydroelectric constraint)} \\
& y_{ijk\omega} \le x_i \qquad \text{for } i = 1, \dots, 4, \text{ all } j, k, \omega \qquad \text{(Capacity constraints)} \\
& \sum_{i=1}^{5} y_{ijk\omega} \ge D_{jk\omega} \qquad \text{for all } j, k, \omega \qquad \text{(Demand constraints)} \\
& x \ge 0, \quad y \ge 0
\end{aligned} \tag{1}
$$

where α_ω is the probability of scenario ω, and $D_{jk\omega}$ is the power demand in block j and year k under scenario ω.

This model has 4 stage-1 variables, and 46,875 stage-2 variables. Furthermore, other than nonnegativity constraints, there are 2 stage-1 constraints, and 375 constraints for each of the 125 scenarios in stage-2. This yields a total of $46,877$ constraints. It is typical for problems of this type to be solved by Benders' decomposition, which exploits the problem structure by decomposing it into smaller problems.

Consider a fixed x which is feasible (i.e., $x \ge 0$ and satisfies the budget constraint and hydroelectric constraint). Then the optimal second-stage variables $y_{ijk\omega}$ can be determined by solving for each ω the problem:

$$
\begin{aligned}
z_\omega(x) \triangleq \min_{y} \quad & \sum_{i=1}^{5}\sum_{j=1}^{5}\sum_{k=1}^{15} 0.01 f_i(\omega) h_j y_{ijk\omega} \\
\text{subject to} \quad & y_{ijk\omega} \le x_i \qquad \text{for } i = 1, \dots, 4, \text{ all } j, k, \qquad \text{(Capacity constraints)} \\
& \sum_{i=1}^{5} y_{ijk\omega} \ge D_{jk\omega} \qquad \text{for all } j, k \qquad \text{(Demand constraints)} \\
& y \ge 0.
\end{aligned} \tag{2}
$$

Although this means solving 125 problems (one for each scenario), each subproblem has only 375 variables and 375 constraints. (Furthermore, the structure of this problem lends itself to a simple greedy optimal solution.)

The dual of the subproblem (2) is:

$$\begin{aligned} z_\omega(x) \triangleq \max_{p,q} \quad & \sum_{i=1}^{4}\sum_{j=1}^{5}\sum_{k=1}^{15} x_i p_{ijk\omega} + \sum_{j=1}^{5}\sum_{k=1}^{15} D_{jk\omega} q_{jk\omega} \\ \text{subject to} \quad & p_{ijk\omega} + q_{jk\omega} \leq 0.01 f_i(\omega) h_j \qquad i = 1, \ldots, 4, \text{ for all } j, k \\ & p_{ijk\omega} \leq 0 \\ & q_{jk\omega} \geq 0. \end{aligned} \tag{3}$$

6 Computational Results

All models were solved in OPLStudio on a Sony Viao Laptop with a Pentium III 750 MHz Processor Running Windows 2000. Table 1 shows the number of iterations taken by the simplex method to solve the original model.

	Original Model Iterations	CPT Time (minutes:seconds)
Simplex	41,334	5:03

Table 1: Iterations to solve the original model.

We next solved the problem using Benders' decomposition, using a duality gap tolerance of $\varepsilon = 10^{-2}$. We first solved the model treating all stage-2 variables and constraints as one large block, using the basic Benders' decomposition algorithm. We next solved the model treating each of the 125 stage-2 scenarios as a separate block, using the more advanced version of Benders' decomposition that takes full advantage of the block-ladder structure. The number of master iterations and the number of constraints generated by the two different solution methods are shown in Table 2. Notice that while the block-ladder version generates more constraints, it requires fewer major iterations. Table 3 shows the computation times of all of the solution methods. From this table, it is evident that the method that takes full advantage of the block-ladder structure was the most efficient solution method.

	Benders' Decomposition	
	Basic (1 Block)	Block-Ladder (125 Blocks)
Master Iterations	39	15
Generated Constraints	39	1,875

Table 2: Master iterations and generated constraints for the two implementations of Benders' decomposition.

	No Decomposition	Benders' Decomposition	
	Original Model	Basic (1 Block)	Block-Ladder (125 Blocks)
CPU Time (min:sec)	5:03	3:22	1:27

Table 3: Computation times (min:sec) for the original model and the two implementations of Benders' decomposition.

Figure 2 shows the the upper and lower bounds generated by Benders' decomposition at each iteration. Notice that there is initial rapid convergence of the bounds to the optimal objective function value, followed by very slow convergence thereafter. This is typical pattern that is observed in decomposition methods in general. Figure 3 shows the duality gap between the upper and lower bounds on a logarithmic scale. Here we see that the convergence rate of the duality gap is roughly linear.

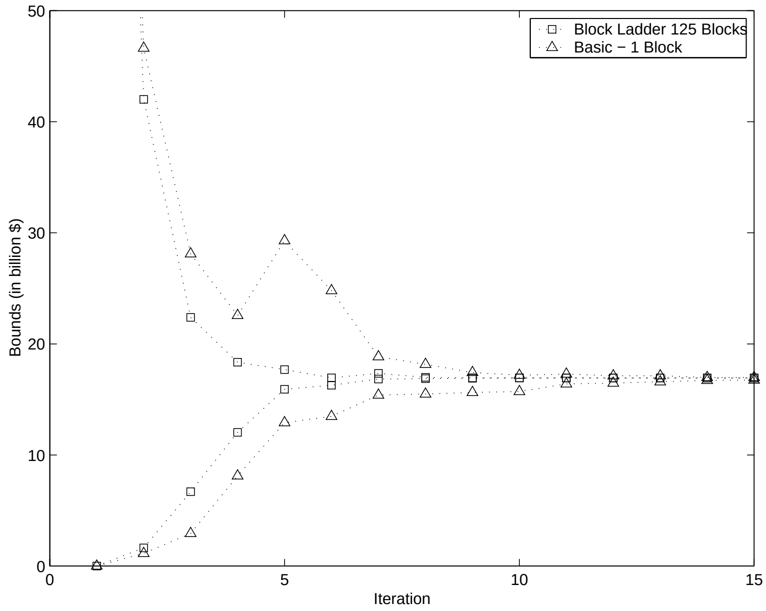

Figure 2: Upper and lower bounds for Benders' decomposition.

7 Exercise

The current version of the powerplant planning model uses the strategy of adding one constraint for every scenario in the model at each iteration of the Benders' decomposition method. This means that $k = 125$ new constraints are added to the model at each outer iteration of the method. As discussed in class, this strategy outperforms the alternative strategy of adding only $k = 1$ new constraint at each outer iteration of the model. In this question, you are asked to explore intermediate strategies that might improve computation time for solving the powerplant planning model using Benders' decomposition.

By adding your own control logic in the region indicated in the script file, `bender1.osc`, experiment with different strategies for limiting and/or controlling the number of new constraints added to the model at each outer iteration. You may also modify the model files or the data file if necessary. (One strategy that you might try would be to add a fixed number of

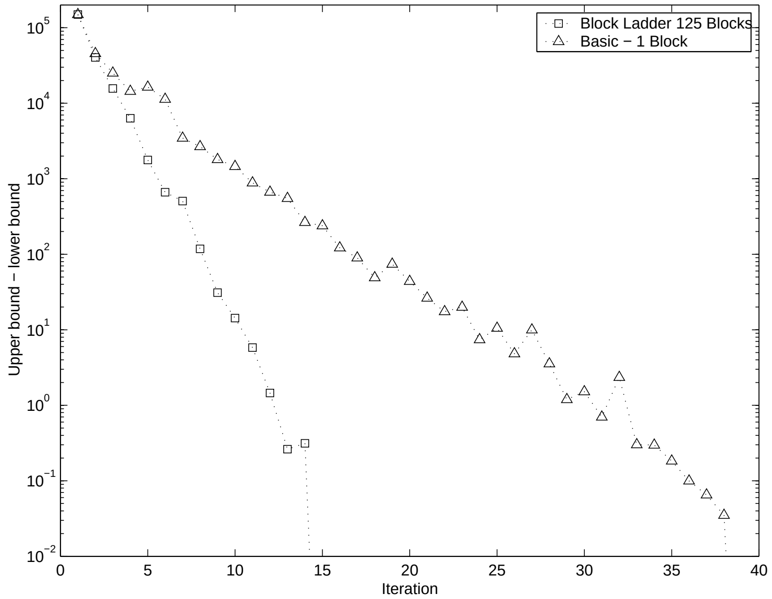

Figure 3: Gap between upper and lower bounds for Benders' decomposition.

constraints, say $k = 10, 20$, or 30, etc., constraints per outer iteration.)

References

[1] D. Anderson. Models for determining least-cost investments in electricity supply. *The Bell Journal of Economics*, 3:267–299, 1972.

[2] J. Benders. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, 4:238–252, 1962.

[3] D. Bertsimas and J. Tsitisklis. *Introduction to Linear Optimization.* Athena Scientific, 1997.

[4] J. Birge and F. Louveaux. *Introduction to Stochastic Programming.* Springer-Verlag, 1997.