Konrad-Zuse-Zentrum für Informationstechnik Berlin
Takustraße 7, D-14195 Berlin-Dahlem, Germany

Claus C. Carøe   Rüdiger Schultz

Dual Decomposition in Stochastic Integer Programming

Preprint SC 96–46 (December 1996)

Dual decomposition in stochastic integer programming

Claus C. Carøe * Rüdiger Schultz †

Abstract

We present an algorithm for solving stochastic integer programming problems with recourse, based on a dual decomposition scheme and Lagrangian relaxation. The approach can be applied to multi-stage problems with mixed-integer variables in each time stage. Numerical experience is presented for some two-stage test problems.

Keywords: Stochastic programming, mixed-integer programming, Lagrangian relaxation, branch-and-bound.

1 Introduction

Stochastic programs with recourse are aimed at finding non-anticipative here-and-now decisions that must be taken prior to knowing the realizations of some random variables such that total expected costs (revenues) from here-and-now decisions and possible *recourse* actions are minimized (maximized). When some of the decision variables are required to be integer or binary we speak of a stochastic integer programming problem.

Stochastic integer programs are challenging from both computational and theoretical points of view since they combine two difficult types of models into one. Until now algorithmic results have been limited to special instances. Laporte & Louveaux [9] developed an integer L-shaped decomposition algorithm for problems with binary first stage and easily computable recourse costs. Løkketangen & Woodruff [10] applied the progressive hedging algorithm and tabu search to multi-stage problems with mixed 0-1 variables. Takriti, Birge and Long [15] report about application of progressive hedging to multi-stage stochastic unit commitment problems in power generation.

*University of Copenhagen, Institute of Mathematics, Universitetsparken 5, DK-2100 Copenhagen, Denmark. E-mail: `caroe@math.ku.dk`. This research was carried out while the author was at Konrad-Zuse-Zentrum für Informationstechnik Berlin.

†Konrad-Zuse-Zentrum für Informationstechnik Berlin (ZIB), Takustraße 7, D-14195 Berlin-Dahlem, Germany. E-mail: `schultz@zib.de` (corresponding author).

A framework for solving two-stage problems using Gröbner bases was proposed by Schultz *et al.* [14], but is limited to problems with integer recourse variables only. For problems with simple integer recourse the expected recourse costs can be made separable in the first stage variables and special methods which are treated in detail in van der Vlerk [16] can be applied.
Decomposition methods for stochastic programs generally fall in two groups: *Primal* methods that work with subproblems assigned to time stages and *dual* methods that work with subproblems assigned to scenarios. Carøe & Tind [3] showed that applying a primal decomposition method like Benders or L-shaped decomposition to (2) in general leads to master problems that are governed by non-convex, non-differentiable functions of the same type as the value function of an integer program. Instead, we will in this paper work with a dual decomposition method combined with branch-and-bound to achieve convergence. This differs from [10, 15] where, although not formally justified, progressive hedging was used for that purpose and proper convergence was empirically observed.

2 Scenario decomposition and Lagrangian relaxation

We consider the following two-stage stochastic program with integer recourse

$$z = \max\{cx + Q(x) \ : \ Ax \le b, \ x \in X\} \tag{1}$$

where $Q(x) = E_\xi \phi(h(\xi) - T(\xi)x)$ and ϕ is given as $\phi(s) = \max\{q(\xi)y \ : \ Wy \le s, \ y \in Y\}$, $s \in \mathbb{R}^{m_2}$. Here c is a known n_1-dimensional vector, A and W are known matrices of size $m_1 \times n_1$ and $m_2 \times n_2$, respectively, and b is a known m_1-vector. The vector ξ is a random variable defined on some probability space $(\Xi, \mathfrak{F}, P)$ and for each $\xi \in \Xi$, the vectors $q(\xi)$ and $h(\xi)$ and the matrix $T(\xi)$ have conformable dimensions. The sets $X \subseteq \mathbb{R}^{n_1}_+$ and $Y \subseteq \mathbb{R}^{n_2}_+$ denote restrictions that require some or all of the variables to be integer or binary. Finally, E_ξ denotes expectation with respect to the distribution of ξ. A *scenario* is a realization of the random variable $(q(\xi), h(\xi), T(\xi))$ corresponding to an elementary event $\xi \in \Xi$. Typically, the distribution of ξ is multivariate. To avoid complications when computing the integral behind E_ξ we assume that we only have a finite number r of scenarios.
In the following we use the notation (q^j, h^j, T^j) for the jth scenario having probability p^j, $j = 1, \ldots, r$. When ξ follows a finite distribution the problem (1) is equivalent to a large, dual block-angular mixed-integer programming problem. Defining for $j = 1, \ldots, r$ the sets

$$S^j := \{(x, y^j) \ : \ Ax \le b, \ x \in X, \ T^j x + W y^j \le h^j, \ y^j \in Y\}$$

the deterministic equivalent can be written

$$z = \max\{cx + \sum_{j=1}^{r} p^j q^j y^j \ : \ (x, y^j) \in S^j, \ j = 1, \dots, r\}. \tag{2}$$

We assume that (2) is solvable, i.e., feasible and bounded. Here feasibility means that there exists a first-stage solution $x \in X$, $Ax \leq b$, such that the feasible regions $\{y^j \in Y \ : \ Wy^j \leq h^j - T^j x\}$, $j = 1, \dots, r$, of all the corresponding second-stage problems are nonempty or, in other words, $Q(x) > -\infty$. Boundedness is achieved by requiring feasibility for the dual to the LP-relaxation of (2).
The fact that (2) is an integer program of block structure leads to decomposition methods to split it into more manageable pieces. In our approach these pieces will correspond to scenario subproblems. The idea in *scenario decomposition* is to introduce copies $x^1, \dots, x^r$ of the first-stage variable x and then rewrite (2) in the form

$$\max\{\sum_{j=1}^{r} p^j (cx^j + q^j y^j) \ : \ (x^j, y^j) \in S^j, \ j = 1, \dots, r, \ x^1 = \dots = x^r\}. \tag{3}$$

Here the *non-anticipativity* conditions $x^1 = \dots = x^r$ state that the first-stage decision should not depend on the scenario which will prevail in the second stage. Of course, there are several equivalent possibilities for expressing this property. Here, we assume that non-anticipativity is represented by the equality $\sum_{j=1}^{r} H^j x^j = 0$ where $H = (H^1, \dots, H^r)$ is a suitable $l \times n_1 r$ matrix.
The Lagrangian relaxation with respect to the non-anticipativity condition is the problem of finding x^j, y^j, $j = 1, \dots, r$ such that

$$D(\lambda) = \max\{\sum_{j=1}^{r} L_j(x^j, y^j, \lambda) \ : \ (x^j, y^j) \in S^j\} \tag{4}$$

where λ has dimension l and $L_j(x^j, y^j, \lambda) = p^j(cx^j + q^j y^j) + \lambda^j (H^j x^j)$ for $j = 1, \dots, r$. The Lagrangian dual of (3) then becomes the problem

$$z_{LD} = \min_{\lambda} D(\lambda). \tag{5}$$

The following weak duality result is well known and can be found in, e.g., Nemhauser & Wolsey [11].

Proposition 1 *The optimal value of the Lagrangian dual* (5) *is an upper bound on the optimal value of* (2). *If for some choice* λ *of Lagrangian multipliers the corresponding solution* (x^j, y^j), $j = 1, \dots, r$ *of the Lagrangian relaxation* (4) *is feasible, then* (x^j, y^j), $j = 1, \dots, r$ *is an optimal solution of problem* (3) *and* λ *is an optimal solution of* (5).

The Lagrangian dual (5) is a convex non-smooth program which can be solved by subgradient methods. A major advantage is that it splits into separate subproblems for each scenario,

$$D(\lambda) = \sum_{j=1}^{r} D_j(\lambda)$$

where

$$D_j(\lambda) = \max\{L_j(x^j, y^j, \lambda) \ : \ (x^j, y^j) \in S^j\}. \tag{6}$$

Each of these r subproblems is a mixed-integer programming problem of size $(m_1 + m_2) \times (n_1 + n_2)$. In contrast, the deterministic equivalent (2) is a mixed-integer program of size $(m_1 + rn_1) \times (m_2 + rn_2)$.
Subgradient methods typically require one function value and one subgradient per iteration as well as a guess of initial multipliers. By convexity, the subgradient $\partial D(\lambda)$ of D at λ is $\partial D(\lambda) = \sum_{j=1}^{r} \partial D_j(\lambda)$ with $\partial D_j(\lambda)$ the set

$$\partial D_j(\lambda) = \text{conv}\,\{\nabla_\lambda L_j(x, y, \lambda) \ : \ (x, y) \text{ solves } (6)\}.$$

Thus $\sum_{j=1}^{r} H^j x^j$ is a subgradient for D where $(x^1, y^1), \ldots (x^r, y^r)$ are optimal solutions of the scenario subproblems (6).
It is well known that due to the integer requirements in (2), solving (5) will give an upper bound on z which in general is larger than z. The next proposition provides some insight into why this duality gap arises.

Proposition 2 *The optimal value z_{LD} of the Lagrangian dual* (5) *equals the optimal value of the linear program*

$$\max\{\sum_{j=1}^{r} p^j(cx^j + q^j y^j) \ : \ (x^j, y^j) \in \text{conv } S^j, \ j = 1, \ldots, r, \ x^1 = \ldots = x^r\} \tag{7}$$

Proof: Theorem 6.2 in [11], p. 327, yields

$$z_{LD} = \max\{\sum_{j=1}^{r} p^j(cx^j + q^j y^j) : (x, y) \in \text{conv } \times_{j=1}^{r} S^j, \ x^1 = \ldots = x^r\}.$$

The assertion then follows from the fact that conv $\times_{j=1}^{r} S^j = \times_{j=1}^{r}$ conv S^j. □
The duality gap occurs because the convex hull of feasible solutions to (3), which is

$$\text{conv}\,\{(x^1, \ldots, x^r, y^1, \ldots, y^r) \ : \ (x^j, y^j) \in S^j, \ j = 1, \ \ldots, r, \ x^1 = \ldots = x^r\},$$

in general is strictly contained in the set of feasible solutions of (7). From Proposition 2 it is also clear that the upper bound on z provided by (5) is not bigger than that obtained by solving the LP-relaxation of (2) which can be written as

$$\max\{\sum_{j=1}^{r} p^j(cx^j + q^j y^j) : (x^j, y^j) \in S^j_{LP}, \ j = 1, \ldots, r, \ x^1 = \ldots = x^r\}$$

where S^j_{LP} arises from S^j by dropping the integer requirements. In fact, our preliminary numerical tests about which we report in Section 5 indicate remarkable improvements of the LP-bound by solving (5).
The gap can only be closed further by grouping together scenarios in larger blocks, thereby enlarging the size of the subproblems (6). For instance, grouping together the ith and jth scenario would yield

$$(\tilde{q}, \tilde{h}, \tilde{T}) = (\left(p^i q^i \ p^j q^j\right), \begin{pmatrix} h^i \\ h^j \end{pmatrix}, \begin{pmatrix} T^i & 0 \\ 0 & T^j \end{pmatrix})$$

having probability $p^i + p^j$ and a corresponding scenario subproblem of size $(m_1 + 2m_2) \times (n_1 + 2n_2)$.
In Birge & Dempster [2] a class of (multi-stage) stochastic programs with integer requirements is presented where the duality gap caused by Lagrangian relaxation of the non-anticipativity conditions vanishes as the number of scenarios tends to infinity. The following formal example shows that for the problems considered in the present paper this cannot be expected in general.

Example 3 *Consider the following two-stage stochastic program with integer recourse where only the right-hand side h is random*

$$\max\{-3x + \sum_{j=1}^{r} p^j 2y^j \ : \ x - \frac{1}{2}y^j \geq h^j,\ 0 \leq x \leq 1,\ y^j \in \{0,1\},\ j = 1, \ldots, r\}. \tag{8}$$

We assume that the number r of scenarios is even and that $p^j = 1/r$ for all $j = 1, \ldots, r$. Let ε^j with $0 < \varepsilon^j < 1/32$, $j = 1, \ldots, r$, be pairwise distinct real numbers. The right hand sides h^j are given by $h^j = \varepsilon^j$ if j is even and $h^j = (1/4) - \varepsilon^j$, otherwise.

We will show that, independent of the number r of scenarios, the mentioned duality gap is bigger than 1/16. To this end we first consider problem (7) for our example. One confirms that $(\frac{1}{2} + \max_j \varepsilon^j, \frac{1}{2}, 1, \ldots, \frac{1}{2}, 1)$ is feasible for that problem such that the maximum is bounded below by $-3 \cdot \max_j \varepsilon^j > -3/32$.
To bound the maximum (8) from above let us first consider the case where $y^j = 1$ for some j that is odd. Then $x \geq (3/4) - \max_j \varepsilon^j$, and the maximum in (8) is bounded above by $-(9/4) + 3 \cdot \max_j \varepsilon^j + 2$ which is less than $-5/32$.
In case $y^j = 0$ for all j that are odd we first consider the situation where $y^j = 0$ for all the remaining j. Then $x \geq (1/4) - \max_j \varepsilon^j$, and the maximum in (8) is bounded above by $-(3/4) + 3 \cdot \max_j \varepsilon^j$ which is less than $-21/32$.
Finally, if $y^j = 0$ for all j that are odd, and $y^j = 1$ for some remaining j, then $x \geq (1/2) + \min_j \varepsilon^j$, and the maximum is bounded above by $-(3/2) - 3 \cdot \min_j \varepsilon^j + 1$ which is less than $-1/2$.
Altogether, the maximum in (8) is thus less than $-5/32$ and, hence, the duality gap is at least 1/16. □

When some of the first-stage decisions in the stochastic program (1) are required to be boolean, then there exists a compound representation of the non-anticipativity constraints. For notational convenience we assume that all first stage decisions are required to be boolean, i.e., that $X = \{0,1\}^{n_1}$. Then we can express non-anticipativity constraint by the single constraint

$$(\sum_{j=2}^{r} a_j)x^1 = a_2x^2 + \cdots a_rx^r \tag{9}$$

where $a_2, \ldots, a_r$ are positive weights. Indeed, the only integer solutions of (9) in $\{0,1\}^{n_1}$ are $(x^1, \ldots, x^r) = (0, \ldots, 0)$ and $(x^1, \ldots, x^r) = (1, \ldots, 1)$ for which the non-anticipativity constraints are satisfied. A special case of (9), see also [2], is obtained by letting $a_j = p^j$ for $j = 1, \ldots, r$ for which (9) can be written

$$(1 - p^1)x^1 = p^2x^2 + \cdots p^rx^r.$$

The Lagrangian with respect to (9) reads

$$\sum_{j=1}^{r} \tilde{L}_j(x^j, y^j, \mu) = \sum_{j=1}^{r} p^j(cx^j + q^jy^j) + \mu \sum_{j=2}^{r} a_j(x^1 - x^j),$$

and the Lagrangian dual

$$z_{LB} = \min_{\mu} \sum_{j=1}^{r} \max\{\tilde{L}_j(x^j, y^j, \mu) \ : \ (x^j, y^j) \in S^j\}. \tag{10}$$

Proposition 4 *The optimal value z_{LB} of the Lagrangian dual* (10) *is greater than or equal to z_{LD}, and z_{LB} equals the optimal value of the problem*

$$\max \left\{ \sum_{j=1}^{r} p^j(cx^j + q^jy^j) : \begin{array}{l} (x^j, y^j) \in \text{conv } S^j, \ j = 1, \ldots, r, \\ (\sum_{j=2}^{r} a_j)x^1 = a_2x^2 + \cdots a_rx^r \end{array} \right\}. \tag{11}$$

Proof: In the same way as in the proof of Proposition 2 it is established that z_{LB} equals the optimal value of (11). Obviously, the feasible region of (11) contains that of (7) which yields $z_{LB} \geq z_{LD}$. □

The advantage of (10) is that the number of Lagrangian multipliers is reduced from l to n_1. The number of multipliers affect subgradient procedures in two ways. A small number of parameters give less controllability and the duality gap is increased, viz. Proposition 4. On the other hand, more control parameters mean that a larger space of parameter settings has to be searched and more iterations may be needed.

Our approach can be related to existing techniques in both combinatorial optimization and in stochastic programming. In combinatorial optimization, the idea of creating copies of variables and then relaxing the equality constraints for these variables was introduced as *variable splitting* by Jörnsten *et al.* [6]. The variable splitting approach was originally applied to optimization problems with a "hard" and an "easy" set of constraints as an alternative to the well-known Lagrangian relaxation approach. The variable splitting method is equivalent to what is called *Lagrangian decomposition* of (2), by Guignard & Kim [5].

In stochastic programming, non-anticipativity conditions are "hard" since they couple constraints for the different scenarios. For linear problems without integer requirements there exist well developed theory and methodology for relaxing non-anticipativity constraints. Based on duality results involving augmented Lagrangians, algorithms like the progressive hedging method of Rockafellar & Wets [12] and the Jacobi method of Rosa & Ruszczyński [13] were developed and applied to a variety of problems.

As elaborated above duality gaps occur in the presence of integer requirements such that the above methods are no longer formally justified. In the next section we will employ branch-and-bound to close the duality gap. This will also lead to optimality estimates for the feasible solutions that are generated in the course of the method.

3 A Branch-and-Bound algorithm

Lagrangian duality provides upper bounds on the optimal value of problem (3) and corresponding optimal solutions (x^j, y^j), $j = 1, \ldots, r$, of the Lagrangian relaxation. In general these *scenario solutions* will not coincide in their x-component unless the duality gap vanishes. We now elaborate a branch-and-bound procedure for (1) that uses Lagrangian relaxation of non-anticipativity constraints as bounding procedure. To come up with candidates for feasible first-stage solutions x various heuristic ideas starting from the scenario solutions x^j, $j = 1, \ldots, r$, can be tried. In the present paper we use the average $\bar{x} = \sum_{j=1}^{r} p^j x^j$, combined with some rounding heuristic in order to fulfill the integrality restrictions. In the following $\mathcal{P}$ denotes the list of current problems and z_i is an upper bound associated with problem $P_i \in \mathcal{P}$. The outline of the algorithm is as follows:

Step 1 *Initialization:* Set $\underline{z} = -\infty$, $z_1 = +\infty$ and let $\mathcal{P} = \{P_1\}$ consist of problem (1).

Step 2 *Termination:* If $\mathcal{P} = \emptyset$ then the solution $\hat{x}$ that yielded $\underline{z} = c\hat{x} + Q(\hat{x})$ is optimal.

Step 3 *Node selection:* Select and delete a problem P_i from $\mathcal{P}$ and solve its Lagrangian relaxation. If the optimal value z_{LD_i} hereof equals $-\infty$ (infeasibility of a subproblem) then go to Step 2.

Step 4 *Bounding:* If $z_{LD_i} \le \underline{z}$ go to Step 2 (this step can be carried out as soon as the value of the Lagrangian dual falls below $\underline{z}$).

(i) The scenario solutions x_i^j, $j = 1, \ldots, r$, are identical: If $cx_i^j + Q(x_i^j) > \underline{z}$ then let $\underline{z} = cx_i^j + Q(x_i^j)$ and delete from $\mathcal{P}$ all problems with $z_i \leq \underline{z}$. Go to Step 2.

(ii) The scenario solutions x_i^j, $j = 1, \ldots, r$ differ: Compute the average $\bar{x}_i$ and round it by some heuristic to obtain $\bar{x}_i^{\mathrm{R}}$. If $c\bar{x}_i^{\mathrm{R}} + Q(\bar{x}_i^{\mathrm{R}}) > \underline{z}$ then let $\underline{z} = c\bar{x}_i^{\mathrm{R}} + Q(\bar{x}_i^{\mathrm{R}})$ and delete from $\mathcal{P}$ all problems with $z_i \leq \underline{z}$. Go to Step 5.

Step 5 *Branching:* Select a component $x_{(k)}$ of x and add two new problems to $\mathcal{P}$ obtained from P_i by adding the constraints $x_{(k)} \leq \lfloor \bar{x}_{i(k)} \rfloor$ and $x_{(k)} \geq \lfloor \bar{x}_{i(k)} \rfloor + 1$, respectively (if $x_{(k)}$ is an integer component) or $x_{(k)} \leq \bar{x}_{i(k)}$ and $x_{(k)} \geq \bar{x}_{i(k)}$, respectively.

In LP-based branch-and-bound algorithms for integer programming upper bounds are obtained by relaxing the integrality requirements and feasibility is obtained when a relaxation has an integer optimum. Here we relax the non-anticipativity requirement and feasibility is obtained when the scenario solutions are identical. With mixed-integer variables, the latter is rarely achieved in early steps of the algorithm. Therefore Step 4(ii) is added where we try to find a feasible solution using the above mentioned rounding. In the best case this might lead to deletion of subproblems from $\mathcal{P}$ and speed up the branch-and-bound procedure. It is convenient to introduce a measure for the dispersion of the components in the scenario solutions, which takes into account different ranges of variables, e.g., boolean and continuous variables. Standard rules for selecting branching variables and nodes can be adapted to our setting using this dispersion measure, for instance by branching on the component $x_{(k)}$ with largest dispersion, selecting the node with the highest norm of dispersion, etc.
In the case of mixed-integer variables some stopping criterion is needed to avoid endless branching on the continuous components. For instance, if we assume that X is bounded and we branch parallel to the coordinate axes, then one may stop after the maximal l_∞-diameter of the feasible sets of the subproblems has fallen below a certain threshold.

Proposition 5 *Suppose that $\{x \in X : Ax \leq b\}$ is bounded and that some stopping criterion for the continuous components is employed. Then the branch-and-bound algorithm above terminates in finitely many steps.*

The implementation of the node selection and branching rules as well as the stopping criterion will depend on the application at hand. Some preliminary experience is presented in Section 5.

4 Extensions to multi-stage stochastic programs

Expanding the two-stage decision process behind the stochastic programs in Section 2 to finite discrete time horizons with arbitrary length leads to multi-stage stochastic programs. In this section we show that, from a formal viewpoint, the above approach can be readily extended to multi-stage stochastic programs with integer requirements. From the implementation point of view, however, some work still remains to be done, since problem sizes increase dramatically. As a prototype example we consider the problem

$$\max\{c_1(\xi_1)x_1 + Q_1(x_1) \ : \ W_1x_1 \leq h_1(\xi_1), \ x_1 \in X_1\} \tag{12}$$

where

$$\begin{aligned} Q_t(x_t) = E_{\xi_{t+1}|\xi_t} \max\{& c_{t+1}(\xi_{t+1})x_{t+1} + Q_{t+1}(x_{t+1}) : \\ & T_{t+1}(\xi_{t+1})x_t + Wx_{t+1} \leq h_{t+1}(\xi_{t+1}), x_{t+1} \in X_{t+1}\} \end{aligned} \tag{13}$$

for $t = 1, \ldots, T-1$ with $Q_T \equiv 0$. Here $\xi = (\xi_1, \ldots \xi_t)$ is a random variable on some probability space $(\Xi, \mathfrak{F}, P)$ and $E_{\xi_{t+1}|\xi_t}$ denotes expectation with respect to the distribution of ξ_{t+1}, conditional on ξ_t. We assume that ξ_1 is known at time $t = 1$. For all realizations of ξ and time stages, $T_t(\xi_t)$ and W_t are $m \times n$ matrices and $h_t(\xi_t)$ and $c_t(\xi_t)$ are vectors in $\mathbb{R}^m$ and $\mathbb{R}^n$, respectively. For convenience we assume that the dimensions are the same for all t. The dynamics of the system as represented in the definition (13) of the expected recourse functions is the simplest possible and can be generalized considerably. Finally, the sets $X_t \subseteq \mathbb{R}^n_+$, $t = 1, \ldots, T$ include integrality restrictions on the decision variables. We assume that the random vector ξ has finite support $\Xi = \{\xi^1, \ldots, \xi^r\}$ and corresponding probabilities $p^1, \ldots, p^r$.

In the multi-stage setting scenarios are realizations corresponding to the elementary events in Ξ of the random variables

$$s(\xi) = (s_1(\xi), \ldots, s_T(\xi))$$

where

$$s_t(\xi) = (c_t(\xi_t), T_t(\xi_t), h_t(\xi_t)), \ t = 1, \ldots, T.$$

With each scenario vector $s(\xi^j)$ we associate a vector of indeterminates $x(\xi^j) = (x_1(\xi^j), \ldots, x_T(\xi^j)), \ j = 1, \ldots, r$.

Problem (12) can now be restated as a large-scale structured mixed-integer program:

$$\begin{aligned} \max \quad & \sum_{j=1}^{r} p^j [c_1(\xi_1^j)x_1(\xi^j) + \cdots c_T(\xi_T^j)x_T(\xi^j)] \\ \text{s.t.} \quad & W_1x_1(\xi^j) \leq h_1(\xi_1^j), \\ & T_{t+1}(\xi_{t+1}^j)x_t(\xi^j) + W_{t+1}x_{t+1}(\xi^j) \leq h_t(\xi_{t+1}^j), \\ & x_t(\xi^j) \in X_t \\ & \text{for} \ \ j = 1, \ldots, r \ \ \text{and} \ \ t = 1, \ldots, T \end{aligned} \tag{14}$$

and the non-anticipativity constraints

$$x_t(\xi^{j_1}) = x_t(\xi^{j_2}) \text{ if } s_\tau(\xi^{j_1}) = s_\tau(\xi^{j_2}) \text{ for all } \tau = 1, \ldots, t. \tag{15}$$

These conditions state that two scenarios with the same history until the tth stage should result in the same decisions until this stage. In other words, decisions are only allowed to depend on the past, not on the future. For problems where uncertainties occur at successive time stages the number of scenarios and thereby the number of variables in (14) grows exponentially with the time horizon. As in the two-stage case we can represent the non-anticipativity conditions (15) by $\sum_{j=1}^r H^j x^j = 0$ with a suitable matrix $H = (H^1, \ldots, H^r)$, now of dimension $l \times nr(T-1)$, if we assume that (15) altogether comprises l equations. We define the Lagrangian for (14) as

$$\sum_{j=1}^r L_j(x^j, \lambda) = \sum_{j=1}^r p^j \sum_{t=1}^T c_t(\xi_t^j) x_t(\xi^j) + \sum_{j=1}^r \lambda^j (H^j x^j)$$

where λ has dimension l, and $L_j(x^j, \lambda) = p^j \sum_{t=1}^T c_t(\xi_t^j) x_t(\xi^j) + \lambda^j (H^j x^j)$. Thus the Lagrangian is again separable with respect to scenarios. The Lagrangian dual is then obtained in the same way as in Section 2. It becomes a convex non-smooth minimization problem with a total of l variables.

Again Lagrangian relaxation of non-anticipativity constraints can be embedded as bounding procedure into a branch-and-bound algorithm as described in Section 3. Since non-anticipativity constraints involve variables from all but the final time stage, branching in the multi-stage case has to concern the variables $x_1, \ldots, x_{T-1}$ instead of only x_1 in the two-stage case. Together with the increased dimension of the Lagrangian dual this more expensive branching is the main source of increased computational effort when extending the scheme from Section 3 to multi-stage models.

5 Numerical examples

We have implemented the branch-and-bound algorithm of Section 3 using NOA 3.0 [7], which is an implementation of Kiwiels proximal bundle method [8] for non-differentiable optimization. The non-anticipativity condition was represented using the constraints $x^1 = x^2$, $x^1 = x^2, \ldots, x^1 = x^r$.

At each node of the branching tree we chose to branch on the component $x_{(k)}$ for which the dispersion $\max_j x_{(k)}^j - \min_j x_{(k)}^j$ was largest. For node selection we chose the node with the largest l_∞-norm of dispersions. To obtain good lower bounds we used this rule intertwined with the Best-Bounds rule. The mixed-integer subproblems were solved using the CPLEX 4.0 Callable Library [4]. The experiments were carried out on a Sun SPARCstation 20 with 160 MB memory.

Example 1: The following stochastic program was adapted from [14]:

$$\max\{\frac{3}{2}x_1 + 4x_2 + Q(x_1, x_2) \ : \ 0 \leq x_1, x_2 \leq 5 \text{ and integer}\} \tag{16}$$

where $Q(x_1, x_2)$ is the expected value of the multiknapsack problem

$$\begin{array}{ll} \max & 16y_1 + 19y_2 + 23y_3 + 28y_4 \\ \text{s.t.} & 2y_1 + 3y_2 + 4y_3 + 5y_4 \leq \xi_1 - x_1, \\ & 6y_1 + y_2 + 3y_3 + 2y_4 \leq \xi_2 - x_2, \quad y_i \in \{0,1\}, \ i = 1, \dots, 4, \end{array}$$

and the random variable $\xi = (\xi_1, \xi_2)$ is uniformly distributed on $\Xi = \{(5,5), (5,5.5), \dots (5,15), (5.5,5), \dots, (15,15)\}$. The deterministic equivalent of (16) is an integer program with 1764 binary variables and 2 integer variables. Attempting to solve the problem with CPLEX yields an optimality gap of more than 25%, stopping enumeration after 500.000 nodes. Using our branch-and-bound algorithm the problem could be solved in 136 seconds CPU-time, yielding the optimal solution $x = (0,4)$ and corresponding value $z = 61.32$. Notice that the scenario subproblems are very small, so better runtimes may be achieved by grouping together scenarios. However, the problem is only meant as a benchmark for testing algorithms.

Example 2: To compare the behavior of our algorithm with problems from the literature having larger second stages, we consider a family of two-stage mixed-integer minimization problems analyzed in [10]. The problems SIZES3, SIZES5 and SIZES10 have 3, 5 and 10 scenarios, respectively, and the scenario subproblems have 10 boolean variables, 65 bounded continuous variables and 31 constraints in each stage, with randomness occurring only in the right-hand side of the second-stage problem.

The computational results are summarized in Table 1. The second column shows the time after which the best feasible solutions were found and the third column shows the lower bounds obtained after 1000 seconds of CPU-time, where the test runs were stopped. Contrary to the method in [10], we can estimate the feasible solutions found to be within 0.2% of the optimum . The Lagrangian dual provides considerably better lower bounds than the LP-relaxation. For our test runs we used 10^{-3} as optimality tolerance in NOA which gave a duality gap at the root nodes of 0.2% – 0.3%. The LP-relaxation, however, gives a duality gap of 2.0% – 2.1%. Notice that a smaller optimality tolerance in NOA will produce better bounds but is also more time consuming.

It should be noted that the size of the duality gap indicates that the scenario solutions have almost identical first-stage components. Hence we have calculated the *value of the stochastic solution* (VSS), see Birge [1], which measures the value of using a stochastic model instead of a deterministic model. For all three problems, the VSS was less than 0.8%, which means that the randomness has little influence on the optimal first-stage solution.

Problem	Best solution	CPU-time	Lower bound
SIZES3	224599.2	127 sec.	224360.0
SIZES5	224680.4	861 sec.	224369.0
SIZES10	224744.3	956 sec.	224311.4

Table 1: Table of results for SIZES-problems

Acknowledgements

We are grateful to Krzysztof C. Kiwiel for permission to use the NOA 3.0 package and to A. Løkketangen and D. L. Woodruff for providing us with data for the SIZES-problems.

References

[1] J. R. Birge, "The value of the stochastic solution in stochastic linear programs with fixed recourse", *Mathematical Programming* **24** 314–325 (1982).

[2] J. R. Birge and M. A. H. Dempster, "Stochastic programming approaches to stochastic scheduling", Technical Report 95 - 12, Department of Industrial and Operations Engineering, The University of Michigan, Ann Arbor, 1995.

[3] C. C. Carøe and J. Tind, "L-shaped decomposition of two-stage stochastic programs with integer recourse", Technical Report, Institute of Mathematics, University of Copenhagen, 1995.

[4] CPLEX Optimization, Inc., *Using the CPLEX Callable Library*, CPLEX Optimization, 1989–1995.

[5] M. Guignard and S. Kim, "Lagrangean decomposition: A model yielding stronger bounds", *Mathematical Programming* **39** 215–228 (1987).

[6] K. O. Jörnsten, M. Näsberg and P. A. Smeds, "Variable splitting–A New Lagrangean Relaxation Approach to Some Mathematical Programming Models", Department of Mathematics Report LiTH-MAT-R-85-04, Linköping Institute of Technology, Sweden, 1985.

[7] K. C. Kiwiel, "User's Guide for NOA 3.0: A Fortran Package for Convex Nondifferentiable Optimization", Systems Research Institute, Polish Academy of Sciences, Newelska 6, 01-447 Warsaw, Poland.

[8] K. C. Kiwiel, "Proximity control in bundle methods for convex nondifferentiable optimization", *Mathematical Programming* **46** 105–122 (1990).

[9] G. Laporte and F. V. Louveaux, "The integer L-shaped method for stochastic integer programs with complete recourse", *Operations Research Letters* **13** 133–142 (1993).

[10] A. Løkketangen and D. L. Woodruff, "Progressive Hedging and Tabu Search Applied To Mixed Integer (0,1) Multi-Stage Stochastic Programming", *Journal of Heuristics*, to appear (1996)

[11] G. L. Nemhauser and L. A. Wolsey, *Integer and Combinatorial Optimization*, Wiley-Interscience, New York, 1988.

[12] R. T. Rockafellar and R. J.-B. Wets, "Scenarios and policy aggregation in optimization under uncertainty", *Mathematics of Operations Research* **16** 1–23 (1991).

[13] C. H. Rosa and A. Ruszczyński, "On augmented Lagrangian decomposition methods for multistage stochastic programs", *Annals of Operations Research* **64** 289–309 (1996)

[14] R. Schultz, L. Stougie and M. H van der Vlerk, "Solving stochastic programs with integer recourse by enumeration: a framework using Gröbner basis reductions" (revised version), submitted to *Mathematical Programming* (1996).

[15] S. Takriti, J. R. Birge, and E. Long "A stochastic model for the unit commitment problem", *IEEE Transactions on Power Systems*, to appear (1996).

[16] M. H. van der Vlerk, "Stochastic Programming with Integer Recourse", Ph.D. Thesis, Institute of Economics, University of Groningen, 1995.