The Journal Mathematical Programming, Series B manuscript No.
(will be inserted by the editor)

A Cross-decomposition Scheme with Integrated Primal-dual Multi-cuts for Two-stage Stochastic Programming Investment Planning Problems

Sumit Mitra · Pablo Garcia-Herreros · Ignacio E. Grossmann

Received: date / Accepted: date

Abstract We describe a decomposition algorithm that combines Benders and scenario-based Lagrangean decomposition for two-stage stochastic programming investment planning problems with complete recourse, where the first-stage variables are mixed-integer and the second-stage variables are continuous. The algorithm is based on the cross-decomposition scheme and fully integrates primal and dual information in terms of primal-dual multi-cuts added to the Benders and the Lagrangean master problems for each scenario. The potential benefits of the cross-decomposition scheme are demonstrated with an illustrative case study for a facility location problem under disruptions, where the underlying LP relaxation is weak, and hence, multi-cut Benders decomposition converges only slowly. If the LP relaxation is improved by adding a tightening constraint, the performance of multi-cut Benders decomposition improves but the cross-decomposition scheme stays competitive and outperforms Benders for the largest problem instance.

Keywords Cross-decomposition · Two-stage stochastic programming · Investment planning

1 Motivation

Two-stage stochastic programming investment planning problems (Birge and Louveaux, 2011) can be hard to solve since the resulting deterministic equivalent programs can lead to very large-scale problems. There are two main approaches, which

I.E. Grossmann
Carnegie Mellon University
Department of Chemical Engineering
5000 Forbes Ave. Pittsburgh, PA 15213
Tel.: +1 (412) 268-3642
Fax: +1 (412) 268-7139
E-mail: grossmann@cmu.edu

can be seen as complementary techniques to address the resulting computational challenge: First, sampling methods (Linderoth et al, 2006) and scenario reduction techniques (Heitsch and Romisch, 2003) can be used to limit the number of required scenarios. Second, decomposition schemes, such as Benders decomposition also known as L-shaped method (Benders, 1962; Geoffrion, 1972; Van Slyke and Wets, 1969) or Lagrangean decomposition (Guignard and Kim, 1987; Caroe and Schultz, 1999) can be applied to exploit decomposable problem structure. In the following we focus on decomposition schemes, which have been successfully applied for solving two-stage stochastic programming problems.

Benders decomposition solves the original problem with iterations between a master problem and subproblems, which are obtained by fixing complicating variables, the investment decisions. Dual information from the subproblems, which describes the sensitivity of the second-stage decisions with respect to the first-stage decisions, is fed back to the master problem, which in turn guides the primal search. The convergence rate of Benders decomposition is good if the underlying relaxation is tight (Magnanti and Wong, 1981; Sahinidis and Grossmann, 1991). However, initially the bound provided by the Benders master problem might be weak and a large number of iterations are potentially required. Therefore, tightening techniques to speed-up the convergence of Benders, such as multi-cuts (Birge and Louveaux, 1988), pareto optimal cuts (Magnanti and Wong, 1981) and cut bundle generation methods (Saharidis et al, 2010) (among others) have been proposed.

Lagrangean decomposition applied to stochastic programming problems is a special form of Lagrangean relaxation, which also decomposes the original problem into scenarios, but relaxes the so-called non-anticipativity constraints that enforce the same investment decisions across scenarios (Caroe and Schultz, 1999). Therefore, Lagrangean decomposition provides a good relaxation since the Lagrangean dual is very similar to the original problem. However, the classical Lagrangean decomposition approach suffers from two weaknesses. First, it might be hard to generate good first-stage feasible solutions from the Lagrangean dual subproblems through heuristics. Second, the update of the multipliers by subgradient optimization (Held and Karp, 1971; Held et al, 1974; Fisher, 1981) or cutting planes (Cheney and Goldstein, 1959; Kelley, 1960) can be a bottleneck that slows down the overall convergence of the algorithm. Techniques to speed up convergence of the multiplier update include (among others) the bundle method (Lemarechal, 1974; Zowe, 1985; Kiwiel, 1990), the volume algorithm (Barahona and Anbil, 2000) and the analytic center cutting plane method (Goffin et al, 1992). There are also strategies to combine bounds from subgradient optimization with cutting planes (Mouret et al, 2011; Oliveira et al, 2013), and to update the multipliers based on dual sensitivity analysis (Tarhan et al, 2013).

However, while there are many papers that focus on improvements for one or the other method, it seems that only few research efforts address combining the complementary strengths of the two decomposition schemes. First proposed by Van Roy (1983), the cross-decomposition algorithm is a framework that unifies Benders and Lagrangean relaxation, which can be seen as duals of each other. The cross-decomposition algorithm iterates between the primal and dual subproblems, where each of the subproblems yields the input for the other one. Convergence tests are used to determine when the iteration between primal and dual problems needs to be augmented with the solution of a primal or dual master problem in order to prevent cycling of the algorithm and restore convergence. Holmberg (1990) generalizes the idea of cross-decomposition and introduces a set of enhanced convergence tests. One of the main ideas in cross-decomposition is to avoid solving the master problems since the solution of these problems, potentially MIPs, is regarded as a hard task. Some variations of the method, e.g. mean value cross-decomposition (Holmberg, 1997a,b) in which a new solution is obtained by averaging over a set of solutions from previous iterations, even completely eliminate the use of master problems at the cost of potentially slow convergence.

To the best of our knowledge, there is almost no work that uses a scheme like cross-decomposition for stochastic programming problems. Interestingly, only 49 results are found by a Google Scholar search on "cross-decomposition" + "stochastic programming" as of January 18^{th} 2014. Thereof, only two publications seem to actively combine Benders and Lagrangean decomposition in a somewhat similar way to our proposed scheme for stochastic programming problems. On the one hand, Cerisola et al (2009) use dual information obtained from a component-based Lagrangean dual relaxation in a nested Benders approach for the unit commitment problem. On the other hand, Sohn et al (2011) derive a mean value cross-decomposition approach for two-stage stochastic programming problems (LP) based on Holmberg's scheme (1992), in which the use of any master problem is eliminated and apply the algorithm to a set of random problem instances. They claim to solve the instances faster than Benders and ordinary cross-decomposition. But from their presentation it is hard to judge to what extent their algorithm might avoid cycling behavior.

At the same time, there are two paradigm shifts that have occurred over the last 20 years that influence the way we might perceive cross-decomposition:

1. **MIP solver improvements and faster computers lead to significantly lower CPU times for MIP problems.**[1] Therefore, there is no longer the need to avoid solving master problems in cross-decomposition schemes. While the Benders (primal) master problems is a potentially hard MIP problem, it needs to be solved anyway in the regular Benders approach. Furthermore, the Lagrangean dual master problem is an LP or QP (with stabilization) if cutting planes are employed. Hence, we would like to solve both master problems to obtain primal and dual guidance on our search. At the same time, we would like to strengthen each master problem with information obtained from both subproblems.
2. **Growing grid computing infrastructure is leading to even more parallelization.** Hence, it is desirable to use these computing resources to compute strong lower as well as upper bounds that rely on parallelization. Benders primal subproblems and Lagrangean dual subproblems are both well-suited for solution in parallel.

Hence, in this paper, we describe an enhanced version of the cross-decomposition scheme for two-stage stochastic investment planning programs with complete recourse, in which first-stage investment decisions are mixed-integer and second-stage

[1] Although, the importance of a good formulation should not be neglected. See Bixby and Rothberg (2007) and Lima and Grossmann (2011) for further details and examples.

decisions are continuous. Once investment decisions are fixed, the problem decomposes into scenarios, which are solved as Benders primal subproblems. The Lagrangean dual subproblem for each scenario is obtained by dualizing non-anticipativity constraints. Both, the primal and the dual subproblem are solved in parallel. The primal search is guided by a multi-cut Benders master problem, which is augmented with optimality cuts obtained from the Lagrangean dual subproblems. The dual search is guided by a disaggregated version of the Lagrangean dual master problem, which is bounded with primal information obtained in previously solved Benders primal subproblems.

2 Problem Statement

We consider a two-stage stochastic programming problem (SP) of the following form:

$$
\begin{aligned}
(SP) \qquad \min \quad TC \quad = \quad & c^T x + \sum_{s\in S} \tau_s d_s^T y_s && (1)\\
\text{s.t.} \qquad & A_0 x \leq b_0 && (2)\\
& A_1 x + B_1 y_s \leq b_1 && (3)\\
& B_s y_s \leq b_s \qquad \forall s \in S && (4)\\
& x \in X && (5)\\
& y_s \geq 0 \qquad \forall s \in S && (6)
\end{aligned}
$$

The objective function (1) minimizes the total expected cost TC, which consists of investment cost ($c^T x$) and expected operational cost ($\sum_{s\in S} \tau_s d_s^T y_s$), where τ_s is the probability for scenario $s \in S$ with $\sum_{s\in S} \tau_s = 1$. The first-stage decisions, x are mixed-integer and correspond to discrete choices for investments and associated capacities:

$$
X = \left\{ x = (x_1, x_2)^T : x_1 \in \{0,1\}^n,\ x_2 \geq 0 \right\} \qquad (7)
$$

All second-stage decisions, y_s, which correspond to operational decisions in scenario s, are continuous. In equation (2), constraints on the investment decisions are specified, e.g. logic constraints on the combination of investments. Equation (3) links the investment decisions with operational decisions, e.g. if capacity is expanded, additional operational decisions are available. In equation (4), operational constraints are specified.

Note that the problem naturally decomposes into scenarios once the investment decisions x are fixed. Furthermore, we can explicitly formulate so-called non-anticipativity constraints, which are derived by duplicating the investment decisions for each scenario (x_s) and enforcing equality constraints across all scenarios. We use the formulation by Caroe and Schultz (1999) and re-write problem (SP) in the following way (SPNAC), where equation (12) represents the non-anticipativity constraints, ($x_1 = x_2 = ... = x_n$), with a suitable matrix $H = (H_1, ..., H_{|S|})$:

$$
\begin{aligned}
(SPNAC) \qquad \min \quad TC \quad &= \quad \sum_{s\in S} \tau_s (c^T x_s + d_s^T y_s) && (8)\\
\text{s.t.} \qquad A_0 x_s \quad &\le \quad b_0 && (9)\\
A_1 x_s + B_1 y_s \quad &\le \quad b_1 && (10)\\
B_s y_s \quad &\le \quad b_s \qquad \forall s \in S && (11)\\
\sum_{s\in S} H_s x_s \quad &= \quad 0 && (12)\\
x_s \in X_s & \qquad \forall s \in S && (13)\\
y_s \ge 0 & \qquad \forall s \in S && (14)
\end{aligned}
$$

In (13), X_s is defined analogously to X:

$$X_s = \left\{ x_s = (x_{1,s}, x_{2,s})^T : x_{1,s} \in \{0,1\}^n,\ x_{2,s} \ge 0 \right\} \qquad \forall s \in S \qquad (15)$$

Corollary 1 *Problems (SP) and (SPNAC) are equivalent.*

Proof Trivially, by substituting non-anticipativity constraints (12). □

3 Ingredients of the Decomposition Algorithm

In the following, we describe the ingredients for our cross-decomposition scheme that is based on Benders and Lagrangean decomposition, which both exploit the decomposable problem structure. For clarity of the presentation, we assume complete recourse, which means that all scenarios are feasible, independent of first-stage decisions. This assumption can be relaxed, but requires an adequate handling of infeasible primal subproblems in the decomposition algorithms as per including primal feasibility cuts and addressing dual unboundedness.

3.1 Subproblems

3.1.1 Benders (Primal) Subproblem

Once the first-stage decisions x are fixed, each scenario s can be solved individually. In order to facilitate full primal-dual integration, it is important that we assign the investment cost in the Benders primal subproblems in the same way we assign the investment cost in the Lagrangean dual subproblem. Therefore, we build our Benders primal subproblems from (SPNAC) but remove the explicit non-anticipativity constraint. If we refer to a given vector of investment decisions $\hat{x}^k$ in iteration k, we assume that it is feasible according to constraints (9) and (12). The Benders subproblem (BSP^k) based on (SPNAC), can be written in two ways for each scenario s: a primal formulation $(BSPp_s^k)$ and a dual formulation $(BSPd_s^k)$, which are equivalent since (BSP_s^k) is an LP that can be solved with zero duality gap. We present both in the following.

$$
\begin{aligned}
(BSPp_s^k) \quad & \min \quad \tau_s(c^T \hat{x}^k + d_s^T y_s) && (16)\\
& \text{s.t.} \quad B_1 y_s \leq b_1 - A_1 \hat{x}^k && (17)\\
& \qquad\quad B_s y_s \leq b_s && (18)\\
& \qquad\quad y_s \geq 0 && (19)
\end{aligned}
$$

Let u_s be the dual multiplier associated with constraint (17) and v_s be the dual multiplier of constraint (18). Then, the dual of $(BSPp_s^k)$ can be written as:

$$
\begin{aligned}
(BSPd_s^k) \quad & \max \quad \tau_s c^T \hat{x}^k + (A_1 \hat{x}^k - b_1)^T u_s - b_s^T v_s && (20)\\
& \text{s.t.} \quad -u_s^T B_1 - v_s^T B_s \leq \tau_s d_s && (21)\\
& \qquad\quad u_s, v_s \geq 0 && (22)
\end{aligned}
$$

Let $\hat{y}_s^k$ be the optimal solution of $(BSPp_s^k)$ and (u_s^k, v_s^k) be the solution of $(BSPd_s^k)$. Let us define $z_{P,s}^{*k}$, in the following way (equivalence due to strong duality):

$$
z_{P,s}^{*k} = \tau_s(c^T \hat{x}^k + d_s^T \hat{y}_s^k) = \tau_s c^T \hat{x}^k + (A_1 \hat{x}^k - b_1)^T u_s^k - b_s^T v_s^k \qquad \forall s \in S,\ k \in K \quad (23)
$$

With $z_P^{*k} = \sum_{s \in S} z_{P,s}^{*k}$, we obtain a valid upper bound on (SPNAC)'s objective function value TC since (BSP^k) is a restriction of (SPNAC):

$$
TC \leq z_P^{*k} \qquad \forall k \in K \quad (24)
$$

3.1.2 Lagrangean (Dual) Subproblem

A valid relaxation of (SPNAC) can be obtained by formulating the Lagrangean dual for (SPNAC), in which the non-anticipativity constraints (12) are dualized in order to apply the framework of Lagrangean decomposition (Guignard and Kim, 1987; Caroe and Schultz, 1999). The Lagrangean Dual (LD^k) is formulated in the following with given fixed Lagrange multipliers μ^k for each scenario s, denoted as (LD_s^k).

$$
\begin{aligned}
(LD_s^k) \quad & \min \quad \sum_{s \in S} \tau_s(c^T x_s + d_s^T y_s) + \mu^k H_s x_s && (25)\\
& \text{s.t.} \quad A_0 x_s \leq b_0 && (26)\\
& \qquad\quad A_1 x_s + B_1 y_s \leq b_1 && (27)\\
& \qquad\quad B_s y_s \leq b_s \qquad \forall s \in S && (28)\\
& \qquad\quad x \in X && (29)\\
& \qquad\quad y_s \geq 0 \qquad \forall s \in S && (30)
\end{aligned}
$$

Let $(\tilde{x}_s^k, \tilde{y}_s^k)$ be the optimal solution of (LD_s^k). Let us define $z_{LD,s}^{*k}$ in the following way:

$$z^{*k}_{LD,s} = \tau_s(c^T \tilde{x}^k_s + d^T_s \tilde{y}^k_s) + \mu^k H_s \tilde{x}^k_s \tag{31}$$

With $z^{*k}_{LD} = \sum_{s\in S} z^{*k}_{LD,s}$, we obtain a valid lower bound on (SPNAC)'s objective function value TC since (LD^k) is a relaxation of (SPNAC):

$$z^{*k}_{LD} \leq TC \qquad \forall k \in K \tag{32}$$

3.1.3 Lagrangean (Primal) Subproblem

The previously described Lagrangean dual subproblem yields a solution, in which some of the original non-anticipativity constraints (12) are most likely violated. Therefore, a primal solution needs to be obtained, which also provides a valid upper bound. In the framework of Lagrangean decomposition, "some heuristic" needs to be applied to generate a first-stage feasible x^k, which is used in (SPNAC) in order to obtain the primal solution. We would like to highlight that the Lagrangean primal subproblem for scenario s is equivalent to the Benders subproblem for scenario s in its primal form $(BSPp_s)$.

3.2 Master problems

3.2.1 Benders (Primal) Master Problem (update of first-stage variables)

Dual optimality cuts from Benders primal subproblems Using the dual solution (u^k_s, v^k_s) from (BSP^k_s), we can derive two types of dual optimality cuts for (SPNAC). The optimality cuts are based on weak duality, which means that the value of any dual feasible solution of the subproblems (BSP^k_s) (across all scenarios s) is a lower bound on the optimal value of the primal (SPNAC) and (u^k_s, v^k_s) stays feasible in $(BSPd^k_s)$ even when $\hat{x}^k$ is replaced by another x. The first cut (33) is the well-known Benders cut (Benders, 1962)

$$TC \geq \sum_{s\in S} \tau_s c^T x + \sum_{s\in S} (A_1 x - b_1)^T u^k_s - \sum_{s\in S} b^T_s v^k_s \qquad \forall k \in K \tag{33}$$

The second cut, presented in equations (34)-(35) is the Benders multi-cut (Birge and Louveaux, 1988), where θ_s is the objective function value for scenario s.

$$TC \geq \sum_{s\in S} \theta_s \tag{34}$$

$$\theta_s \geq \tau_s c^T x + (A_1 x - b_1)^T u^k_s - b^T_s v^k_s \qquad \forall s \in S,\ \ k \in K \tag{35}$$

While the multi-cut version (34)-(35) provides faster convergence due to increased strength of the dual information, it increases the problem size compared to the single-cut version (33). Note that we do not provide feasibility cuts since we assume complete recourse, as previously stated.

Lagrangean dual bounds for the Benders primal master problem We notice that the Lagrangean dual (LD_s) is a relaxation of the primal problem for scenario s. Hence, we can derive a valid lower bound for θ_s based on the previously obtained information from the Lagrangean dual:

$$z^{*k}_{LD,s} \leq \theta_s + \mu^k H_s x \qquad \forall s \in S, \;\; k \in K. \tag{36}$$

The proof that (36) is a valid inequality for the Benders primal master problem can be found in Appendix A.1.

No-Good Cut If the complicating variables, x, are all binary variables, we can also formulate a no-good cut (Balas and Jeroslow, 1972) to exclude previously visited solutions from the solution space:

$$\sum_{l \in L_1^k} x_l - \sum_{l \in L_0^k} x_l - |L_1^k| + 1 \leq 0 \qquad \forall k \in K \tag{37}$$

In equation (37), L_0^k and L_1^k are defined as follows (where x_l is the l^{th} component of x):

$$L_0^k = \{l : x_l = 0 \text{ in iteration } k\} \tag{38}$$

$$L_1^k = \{l : x_l = 1 \text{ in iteration } k\} \tag{39}$$

Formulation of the Benders (primal) master problem Using the multi-cut version (34)-(35) for the optimality cuts and the Lagrangean dual bounds (36) presented in (43) below, we write the Benders master problem (BMP^{k+1}) that yields the next primal vector $\hat{x}^{k+1}$ as follows:

$$(BMP^{k+1}) \quad \min \;\; \eta_{BMP} \tag{40}$$

$$\text{s.t.} \;\; \eta_{BMP} \geq \sum_{s \in S} \theta_s \tag{41}$$

$$\theta_s \geq \tau_s^k c^T x + (A_1 x - b_1)^T u_s^k - b_s^T v_s^k \qquad \forall s \in S, \;\; k \in K \tag{42}$$

$$\theta_s \geq z^{*k}_{LD,s} - \mu^k H_s x \qquad \forall s \in S, \;\; k \in K \tag{43}$$

$$A_0 x \leq b_0 \tag{44}$$

$$x \in X, \;\; \eta_{BMP} \in \mathbb{R}^1, \;\; \theta_s \in \mathbb{R}^1 \qquad \forall s \in S \tag{45}$$

If X has only integer variables, we can add the no-good cut (37) with the sets L_0^k and L_1^k defined by (38)-(39) to (BMP^{k+1}). Let z^{*k+1}_{BMP} be the optimal objective function value of (BMP^{k+1}). Note that (BMP^{k+1}) is a relaxation of (SPNAC) that provides a lower bound to (SPNAC):

$$z^{*k+1}_{BMP} \leq TC \qquad \forall k \in K \tag{46}$$

Note that by adding equation (43) to the Benders master problem, we strengthen its formulation, and we can guarantee that the lower bound obtained from the Benders master problem is at least as tight as the best known solution from the Lagrangean dual since

$$z_{LD}^{*k'} = \sum_{s\in S} z_{LD,s}^{*k'} - \mu^{k'} \underbrace{\sum_{s\in S} H_s x}_{=0} \le \sum_{s\in S} \theta_s \le z_{BMP}^{*k+1} \qquad \forall\ k\in K,\ k'\in K,\ k'\le k. \quad (47)$$

3.2.2 Lagrangean (Dual) Master Problem (multiplier update)

Once the Lagrangean dual subproblems are solved, a new solution for the Lagrangean multipliers μ needs to be generated for the next iteration of the algorithm based on the information obtained in the subproblems. The most commonly used method to update the Lagrangean multipliers is the subgradient method (Held and Karp, 1971; Held et al, 1974; Fisher, 1981). Unfortunately, the convergence of the subgradient method is not very reliable, especially when dealing with large-scale problems. One alternative to the subgradient algorithm are cutting planes, similar to the ones used in the Benders master problem. We present in the following the Lagrangean dual master problem, which is also referred to as ”cutting plane method” and can be formally derived from the Dantzig-Wolfe primal master problem (Guignard, 2003; Frangioni, 2005).

Typical cutting plane The typical form of the cutting plane (Cheney and Goldstein, 1959; Kelley, 1960) is as follows:

$$TC \le \sum_{s\in S} \tau_s (c^T \tilde{x}_s^k + d_s^T \tilde{y}_s^k) + \mu \sum_{s\in S} H_s \tilde{x}_s^k \qquad \forall k\in K, \quad (48)$$

in which $(\tilde{x}_s^k, \tilde{y}_s^k)$ is the solution of the k^{th} Lagrangean dual subproblem of scenario s (LD_s^k).

Disaggregated version of the cutting plane (multi-cut) In many publications that employ a cutting plane approach for Lagrangean decomposition, the cutting plane as described in equation (48) is used. However, it is also possible to derive a disaggregated version of the cutting plane for the case in which the Lagrangean relaxation decomposes into a set of independent subproblems (Guignard, 2003; Frangioni, 2005), in which κ_s is the variable associated with the objective function value in scenario s:

$$TC \le \sum_{s\in S} \kappa_s \quad (49)$$

$$\kappa_s \le \tau_s (c^T \tilde{x}_s^k + d_s^T \tilde{y}_s^k) + \mu H_s \tilde{x}_s^k \qquad \forall s\in S,\ k\in K \quad (50)$$

The disaggregated version of the cutting plane is analogous to the multi-cut version of the Benders cut, which seems intuitive since Benders decomposition and Lagrangean relaxation can be seen as duals of each other. It can also be shown that (50) is an inner approximation of the convex hull of the non-complicating constraints for scenario s. Therefore, we obtain a tighter approximation of the convex hull of all non-complicating constraints by intersecting these approximations (50) across all scenarios s instead of building the approximation of the convex hull for the entire set of non-complicating constraints (Frangioni, 2005).

Note also that similarly as for the Benders multi-cut, there might be a tradeoff between tightness of the cuts and problem size.

Benders primal bounds for the Lagrangean master problem Analogously to (36), it is further possible to derive additional bounds from the Benders primal subproblems for the objective function value κ_s of each scenario s in the Lagrangean master problem since we will use the disaggregated version of the Lagrangean master problem:

$$\kappa_s \leq z_{P,s}^{*k} + \mu H_s \hat{x}^k \qquad \forall s \in S,\ k \in K \tag{51}$$

The proof that (51) is a valid inequality for the Lagrangean dual master problem is given in Appendix A.2.

Formulation of the Lagrangean Dual Master Problem Using the multi-cut version (49)-(50) of the cutting planes and the upper bounds from the Benders primal subproblem (51) presented in (55) below, the Lagrangean dual master problem (LMP^{k+1}), which yields the next Lagrangean multipliers μ^{k+1}, can be formulated as:

$$(LMP^{k+1}) \quad \max \ \eta_{LMP} + \frac{\delta}{2}\|\mu - \bar{\mu}\|_2^2 \tag{52}$$

$$\text{s.t.} \ \eta_{LMP} \leq \sum_{s \in S} \kappa_s \tag{53}$$

$$\kappa_s \leq \tau_s(c^T \tilde{x}_s^k + d_s^T \tilde{y}_s^k) + \mu H_s \tilde{x}_s^k \qquad \forall s \in S,\ k \in K \tag{54}$$

$$\kappa_s \leq z_{P,s}^{*k} + \mu H_s \hat{x}^k \qquad \forall s \in S,\ k \in K \tag{55}$$

$$\eta_{LMP} \in \mathbb{R}^1,\ \mu \in \mathbb{R}^{(|S|-1)\times n},\ \kappa_s \in \mathbb{R}^1 \qquad \forall s \in S \tag{56}$$

where $|S|$ is the number of scenarios and n the number of dualized first-stage variables.

The objective function (52) contains the additional quadratic stabilization term $\frac{\delta}{2}\|\mu - \bar{\mu}\|_2^2$ that defines a trust-region for the update of the Lagrangean multipliers (Lemarechal, 1974; Zowe, 1985; Kiwiel, 1990; Frangioni, 2002). The stabilization requires initial values and update strategies for the penalty δ, which defines the size of the trust region, and the stabilization center $\bar{\mu}$. For a detailed overview on the stabilization techniques, we refer to the previously mentioned work.

Let z_{LMP}^{*k+1} be the optimal objective function value of (LMP^{k+1}). Note that the valid inequalities (55) make the Lagrangean master problem bounded. Furthermore, they guarantee that the Lagrangean master problem will yield a bound at least as tight as the best known primal upper bound, obtained from the Benders subproblems:

$$z_{LMP}^{*k+1} \leq \sum_{s \in S} \kappa_s \leq \sum_{s \in S} z_{P,s}^{*k'} + \mu \underbrace{\sum_{s \in S} H_s \hat{x}^{k'}}_{=0} = z_P^{*k'} \qquad \forall k \in K,\ k' \in K,\ k' \leq k. \tag{57}$$

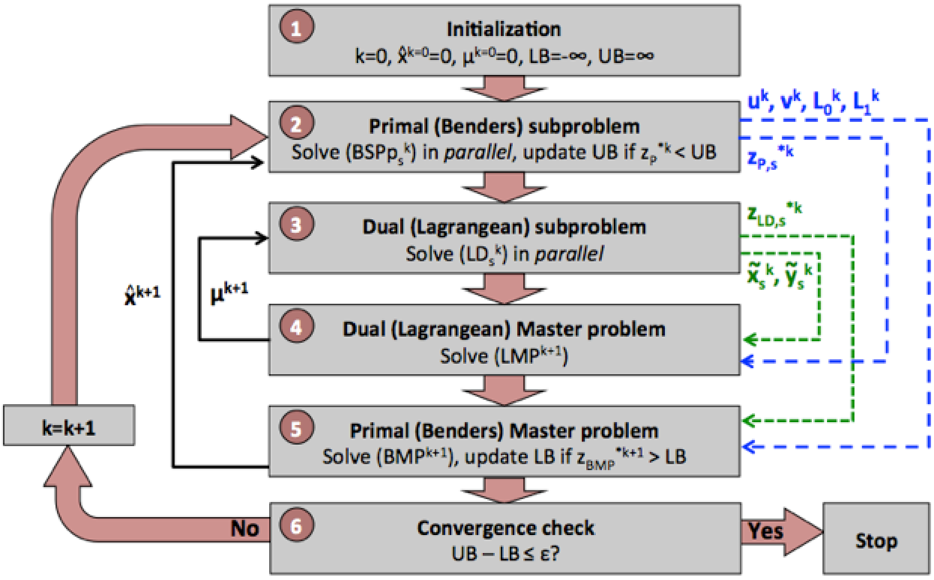

Fig. 1 Algorithmic flow (bold arrows) and flow of information (thin arrows) for our cross-decomposition scheme with multi-cuts.

3.3 Initialization scheme

In Benders decomposition as well as in Lagrangean decomposition, an initial guess for the first-stage primal variables / Lagrangean multipliers needs to be provided. The solution of (SPNAC)'s LP relaxation is usually hard due to a large number of scenarios. We notice that a good guess might speed up convergence and can potentially be derived based on in-depth problem knowledge. A generic initialization is the following one:

$$\hat{x}^{k=0} = 0 \tag{58}$$

$$\mu^{k=0} = 0 \tag{59}$$

In general, this initialization is reasonable since it corresponds to making no investments (as a baseline guess) and assuming that the value transferred to each scenario by making an investment is zero. Note that this initialization will be primal feasible since we have complete recourse. Furthermore, initial values for the trust-region penalty δ and for the stabilization center $\bar{\mu}$ need to be specified. Typically, $\bar{\mu}$ is set to $\mu^{k=0}$.

4 The proposed algorithm

The proposed algorithm, which is illustrated in Fig. 1, consist of the following steps:

1. **Initialization**
   $k = 0$, K=$\{0\}$, $\hat{x}^{k=0} = 0$, $\mu^{k=0} = 0$, $LB = -\infty$, $UB = \infty$, set $\varepsilon \geq 0$,
   Go to step 2
2. **Benders primal subproblem**
   For given $\hat{x}^k$, solve $(BSPp_s^k)$, $\forall s \in S$ in parallel
   Store (u^k, v^k), and generate sets L_0^k, L_1^k if X has only binary variables
   If $z_P^{*k} < UB$, set $UB = z_P^{*k}$

3. **Lagrangean dual subproblem**
   For given μ^k, solve (LD_s^k), $\forall s \in S$ in parallel
   Store solution $\tilde{x}_s^k$, $\tilde{y}_s^k$ and $z_{LD,s}^{*k}$
   Go to step 4
4. **Lagrangean (Dual) Master problem**
   For given $\tilde{x}_s^k$, $\tilde{y}_s^k$, $z_{P,s}^{*k}$, $\forall s \in S,\ k \in K$, solve (LMP^{k+1})
   Store solution for Lagrangean multipliers μ and set them as μ^{k+1}
   Update δ and $\bar{\mu}$ according to update strategies
   Go to step 5
5. **Benders (Primal) Master problem**
   For given (u^k, v^k), (L_0^k,L_1^k), $z_{LD,s}^{*k}$, $\forall s \in S,\ k \in K$, solve (BMP^{k+1})
   Store solution for first-stage primal investment decisions x and set them as $\hat{x}^{k+1}$
   Set $LB = z_{BMP}^{*k+1}$
   Go to step 6
6. **Check Convergence (Optimality)**
   If $UB - LB \leq \varepsilon$: Stop.
   Else set $k = k+1$ and include in K; go back to Step 2.

5 Discussion

5.1 Comments on the primal and dual bounds

One of the potential strengths of the proposed algorithm is that it guarantees to provide a lower bound, which is at least as tight as the best lower bound obtained from the Lagrangean dual, while having a mechanism to generate first-stage solutions that are feasible regarding the first-stage constraints (in contrast to using a heuristic as in Lagrangean decomposition). Note that primal-dual information exchange in terms of bounds for each scenario s is facilitated by the fact that both subproblems, the Benders primal and the Lagrangean dual, have the same investment cost terms for each scenario s.

Furthermore, we use a multi-cut approach for both, the primal Benders and dual Lagrangean master problem to derive primal and dual bounds for each scenario s. While we expect tight bounds, the computational performance needs to be carefully investigated for each case individually. We anticipate to see an impact for cases in which the Benders algorithm converges slowly due to a weak underlying linear relaxation. At the same time, the effort that an additional Lagrangean iteration adds to the total computational time needs to be compared with the number of saved Benders iterations and associated computational time.

5.2 Comments on the original cross-decomposition

In the original cross-decomposition algorithm (Van Roy, 1983; Holmberg, 1990) the master problems are seen as a fall-back option in case one of the convergence tests fails. Otherwise, the solution of the master problems is avoided as much as possible. Therefore, no strengthening techniques such as multi-cuts or valid inequalities (i.e. valid inequalitites (36) and (57)) are introduced.

Note that the convergence tests used in the original cross-decomposition scheme (Van Roy, 1983; Holmberg, 1990) are no longer needed since we solve both master problems in each iteration and therefore can guarantee convergence as proved in Appendix A.3. Furthermore, Holmberg (1990) does not explicitly consider the case of a decomposable structure and associated difficulties in the exchange of primal and dual variables.

6 Illustrative Case Study

In order to provide some insight into the potential of the proposed algorithm, we apply the cross-decomposition scheme to a facility location problem with distribution centers (DCs) under the risk of disruptions. In the problem, demand points need to be satisfied from a set of candidate DCs in order to minimize the sum of investment cost and expected transportation cost. The so-called capacitated reliable facility location problem (CRFLP) is formulated as a two-stage stochastic program. First-stage decision involve the selection of DCs and their capacities. Second-stage decisions are the demand assignments to DCs in scenarios, which are defined by combinations of active and disrupted locations. Penalties are applied to unsatisfied demands, such that the second-stage subproblems have full recourse.

Two versions of the model are presented in Appendix B: (CRFLP) and (CRFLP-t). The second model (CRFLP-t) includes a redundant set of constraints that improves its linear relaxation. The equations of model (CRFLP) and (CRFLP-t) have the same structure as the generic ones described in formulation (1)-(6). Due to its large size, the problem is hard to solve with a fullspace approach and requires decomposition techniques to enable an efficient solution. Furthermore, the problem is numerically challenging because the scenarios with several simultaneous disruptions have very small probabilities.

6.1 Description of implementation

We implement the fullspace model, the classical multi-cut Benders decomposition (Birge and Louveaux, 1988) and our cross-decomposition algorithm, as previously described in section 4, in GAMS 24.1.2 (Brooke et al, 2013).

For the decomposition schemes, we employ parallel computing in two different ways: First, Benders and Lagrangean subproblems are solved in parallel as groups of 50 scenarios using the GAMS grid computing capabilities (Bussieck et al, 2009) on the 8 processors of an Intel i7-2600 (3.40 GHz) machine with 8 GB RAM. We choose to solve the subproblems as groups of scenarios to reduce the overhead in data transfer. Second, the Lagrangean and Benders master problems are solved by allowing GUROBI to use the processors as parallel threads. For the fullspace implementation, we also use GUROBI with parallel threads.

To cope with the numerical difficulties that arise due to small scenario probabilities, we use additional solver settings of GUROBI 5.5. We set `quad 1` (for quad precision) and `numericfocus 3` (for high attention to numerical issues) in all sub- and master problems. The optimality tolerances for reduced costs are set to 10^{-7}. For the Lagrangean master problem, we use `barhomogeneous 1` to detect infeasibilities with the barrier method employed to solve the QPs of the Lagrangean master problem. Additionally, all MILPs are solved to optimality (with respect to the numerical tolerances) with the `optcr=0` setting in GAMS. Benders decomposition and the cross-decomposition algorithm are terminated when the relative difference between the upper and lower bound is less than 10^{-7}.

For the Lagrangean master problem, the multipliers are bounded below and above according to their interpretation, the maximum penalties that can be incurred in a scenario. Additionally, the Lagrange multipliers are scaled with their corresponding scenario probabilities in order to maintain the same order of magnitude among the variables of the problem.

The update strategies for the penalty term δ (initial value $\delta = 1$) and for the stability center $\bar{\mu}$ are rather simple for our illustrative case study. In each iteration, the trust-region parameter δ is updated according the following rule: $\delta^{k+1} = \max\{\frac{1}{2}\delta^k, 10^{-10}\}$. The initial stability center $\bar{\mu} = 0$ is never updated. Note that more elaborate update schemes, as described by Kiwiel (1990) and Frangioni (2002), could additionally improve the strength of the Lagrangean cuts, and hence the performance of the overall cross-decomposition algorithm.

6.2 Computational Results

In the following, we compare the three methods (fullspace model, multi-cut Benders and cross-decomposition algorithm) for different instances of the two problem formulations, (CRFLP) and (CRFLP-t), presented in Appendix B.

6.2.1 Comparison of methods for (CRFLP)

In Table 1, we report the problem sizes of the fullspace model for (CRFLP) for 3 instances with different number of candidate locations for DCs. With an increase in the number of DCs, the number of scenarios, whose generation is described in Appendix B, and the number of constraints and variables increases as well. The resulting optimization problems are large-scale, especially in terms of continuous variables. Note that the number of binary variables is not as large since only one binary variable per DC is required.

Table 1 Sizes of the resulting fullspace optimization problems for (CRFLP) in terms of constraints and variables.

DCs (N)	Scenarios	Constraints	Variables	Binary Var.
10	639	38,992	345,084	10
11	1025	63,564	603,751	11
12	1587	99,996	1,012,534	12

As we can see in Table 2, the resulting fullspace models are hard to solve. While all of them can be solved to optimality, it takes 722 minutes (more than 12 hours) to solve the largest instance with 12 candidate DCs.

Interestingly, multi-cut Benders decomposition fails in solving the problem instances faster than the fullspace method. Only for the smallest instance with 10 candidate DCs, Benders decomposition obtains the optimal solution, which takes 2,871 minutes (almost two days). For the larger instances with 11 and 12 candidate DCs, the algorithm is terminated after a time limit of 4,000 minutes, with a remaining optimality gap of 12.2% and 18.8%, respectively. In these cases, the optimality gap does not close after a large number of iterations (239 and 130, respectively).

In contrast, the cross-decomposition algorithm solves all three instances to optimality, and outperforms the fullspace model in terms of runtime by 48% and 34% for 10 and 12 candidate DCs, respectively. For 11 candidate DCs, the fullspace model and the cross-decomposition algorithm require nearly the same amount of time.

As it was shown by Magnanti and Wong (1981) as well as Sahinidis and Grossmann (1991), the performance of Benders decomposition depends on the strength of the underlying LP relaxation. From Table 2, one can observe that the LP relaxation provides poor bounds, which explains the bad performance of the multi-cut Benders decomposition. Hence, we can make the preliminary conclusion that cross-decomposition is less affected by the weak LP relaxation due to the presence of strong cuts that originate from the Lagrangean dual.

6.2.2 Effect of tightening constraint: comparison of methods for (CRFLP-t)

As presented in Appendix B, a tightening constraint that improves the LP relaxation of (CRFLP) is known. However, the tightening constraint (86) significantly increases the problem size for the fullspace model, denoted by (CRFLP-t), as one can see in Table 3.

In the following, we discuss the computational impact of constraint (86) and the resulting model (CRFLP-t). As one can see in Table 4, the solution time of the fullspace model for (CRFLP-t) increases significantly compared to the solution time of (CRFLP)'s fullspace model (Table 2). At the same time, the LP relaxations for all instances of (CRFLP-t) are much tighter than (CRFLP)'s LP relaxation. Hence,

Table 2 Computational results for (CRFLP)

DCs (N)		**Fullspace**	**Benders**	**Cross**
10	Objective ($)	1,003,707.23	1,003,707.23	1,003,707.23
	LP relaxation ($)	520,311.87	-	-
	Optimality gap (%)	0	0	0
	Iterations (#)	-	440	21
	Runtime (min)	33	2871	17
11	Objective ($)	1,003,632.26	1,007,279.28	1,003,632.26
	LP relaxation ($)	495,055.02	-	-
	Optimality gap (%)	0	12.2	0
	Iterations (#)	-	239	37
	Runtime (min)	167	4000[a]	182
12	Objective ($)	1,004,855.83	1,028,650.66	1,004,855.83
	LP relaxation ($)	479,563.62	-	-
	Optimality gap (%)	0	18.8	0
	Iterations (#)	-	130	36
	Runtime (min)	722	4,000[a]	474

[a]: terminated after 4000 min time limit.

Table 3 Sizes of the resulting fullspace optimization problems (CRFLP-t) (with tightening constraint).

DCs (N)	Scenarios	Constraints	Variables	Binary Var.
10	639	383,413	345,084	10
11	1025	666,264	603,751	11
12	1587	1,110,915	1,012,534	12

multi-cut Benders decomposition solves all instances faster than the fullspace model (speed-up of 78% and more in terms of runtime) since stronger cuts are generated in the subproblems.

In comparison with Benders decomposition, cross-decomposition always takes fewer iterations due to its strong Lagrangean bounds. However, the gains in terms of iterations and stronger lower bounds come with the price of additional computational time that is required to solve the Lagrangean dual subproblems and the Lagrangean master problem in each iteration. For 10 and 11 candidate DCs, cross-decomposition is 4 and 2 minutes slower than Benders decomposition, respectively.

As the problem size increases, the strong bounds obtained by the Lagrangean part of the cross-decomposition become more powerful and have a bigger impact. For the largest instance (12 candidate DCs), cross-decomposition achieves a reduction around 63% in both runtime and number of iterations compared to Benders decomposition.

In Fig. 2, the convergence of the lower and upper bounds for Benders decomposition and cross-decomposition is shown for the largest instance (12 candidate DCs) of the (CRFLP-t). While Benders decomposition takes 65 iterations, the cross-

Table 4 Computational results for (CRFLP-t).

DCs (N)		**Fullspace**	**Benders**	**Cross**
10	Objective ($)	1,003,707.231	1,003,707.231	1,003,707.231
	LP relaxation ($)	1,000,314.99	-	-
	Optimality gap (%)	0	0	0
	Iterations (#)	-	13	12
	Runtime (min)	61	2.5	6.5
11	Objective ($)	1,003,632.26	1,003,632.26	1,003,632.26
	LP relaxation ($)	995,531.18	-	-
	Optimality gap (%)	0	0	0
	Iterations (#)	-	38	20
	Runtime (min)	328	42	44
12	Objective ($)	1,004,855.83	1,004,855.83	1,004,855.83
	LP relaxation ($)	996,777.36	-	-
	Optimality gap (%)	0	0	0
	Iterations (#)	-	65	24
	Runtime (min)	2,015	435	158

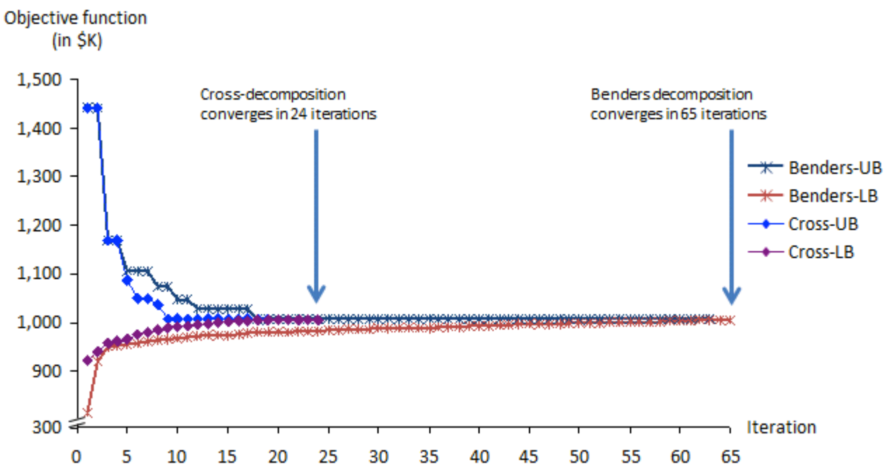

Fig. 2 Convergence of multi-cut Benders decomposition and cross-decomposition for instance with 12 candidate DCs of the (CRFLP-t).

decomposition scheme requires only 24 iterations. One can clearly see that the lower bounds obtained from the cross-decomposition scheme always dominate the lower bounds obtained from Benders decomposition.

7 Conclusion

We have described a cross-decomposition algorithm that combines Benders and scenario-based Lagrangean decomposition for two-stage stochastic MILP problems with complete recourse, where the first-stage variables are mixed-integer and the second-stage variables are continuous. The algorithm fully integrates primal and dual information with multi-cuts that are added to the Benders and the Lagrangean master problems for each scenario. Computational results for an illustrative case study on a facility location problem under the risk of disruptions show evidence of the conceptual strength of the cross-decomposition such as a reduction of iterations and stronger lower bounds compared to pure multi-cut Benders decomposition. While the computational times per iteration increases due to the solution of the Lagrangean dual subproblem and master problem, cross-decomposition seems to be especially advantageous compared to Benders decomposition if the underlying LP relaxation is weak, as suggested by the facility location problem. Despite the promising potential of the cross-decomposition scheme, more computational experiments should be conducted in order to fully assess its advantages in different types of problems.

Acknowledgements We would like to thank the National Science Foundation for financial support under grant # CBET-1159443.

References

Balas E, Jeroslow RG (1972) Canonical Cuts on the Unit Hypercube. SIAM Journal of Applied Mathematics 23:61–69

Barahona F, Anbil R (2000) The Volume Algorithm: Producing Primal Solutions with a Subgradient Method. Mathematical Programming 87:385–399

Benders JF (1962) Partitioning Procedures for Solving Mixed-variables Programming Problems. Numerische Mathematik 4:238–252

Birge JR, Louveaux F (1988) A Multicut Algorithm for Two-stage Stochastic Linear Programs. European Journal of Operational Research 34:384–392

Birge JR, Louveaux F (2011) Introduction to Stochastic Programming. Springer Series in Operations Research, Springer

Bixby R, Rothberg E (2007) Progress in Computational Mixed Integer Programming - a Look back from the other Side of the Tipping Point. Annals of Operations Research 49:37–41

Brooke A, Kendrick D, Meeraus A (2013) GAMS: A Users Guide, Release 24.2.1. The Scientific Press, South San Francisco

Bussieck MR, Ferris MC, Meeraus A (2009) Grid-Enabled Optimization with GAMS. INFORMS Journal on Computing 21:349–362

Caroe CC, Schultz R (1999) Dual Decomposition in Stochastic Integer Programming. Operations Research Letters 24:37–45

Cerisola S, Baillo A, Fernandez-Lopez JM, Ramos A, Gollmer R (2009) Stochastic Power Generation Unit Commitment in Electricity Markets: A Novel Formulation and a Comparison of Solution Methods. Operations Research 57:32–46

Cheney EW, Goldstein AA (1959) Newton's Method for Convex Programming and Tchebycheff Approximation. Numerische Mathematik 1:253–268
Daskin MS (1995) Network and Discrete Location: Models, Algorithms, and Applications, John Wiley & Sons, Inc, chap Appendix H. Longitudes, Latitudes, Demands, and Fixed Cost for SORTCAP.GRT: A 49-Node Problem Defined on the Continental United States
Fisher ML (1981) The Lagrangian Relaxation Method for Solving Integer Programming Problems. Management Science 27:1–18
Frangioni A (2002) Generalized Bundle Methods. SIAM Journal on Optimization 13:117–156
Frangioni A (2005) About Langrangian Methods in Integer Optimization. Annals of Operations Research 139:163–193
Garcia-Herreros P, Wassick J, Grossmann IE (2014) Design of Resilient Supply Chains with Risk of Facility Disruptions. in preparation
Geoffrion AM (1972) Generalized Benders Decomposition. Journal of Optimization Theory and Applications 10:237–260
Goffin JL, Haurie A, Vial JP (1992) Decomposition and nondifferentiable optimization with the projective algorithm. Management Science 38:284–302
Guignard M (2003) Lagrangean Relaxation. Top 11:151–228
Guignard M, Kim S (1987) Lagrangean Decomposition - a Model Yielding Stronger Lagrangean Bounds. Mathematical Programming 39:215–228
Heitsch H, Romisch W (2003) Scenario Reduction Algorithms in Stochastic Programming. Computational Optimization and Applications 24:187–206
Held M, Karp RM (1971) The Traveling-Salesman Problem and Minimum Spanning Trees: Part ii. Mathematical Programming 1:6–25
Held M, Wolfe P, Crowder HP (1974) Validation of Subgradient Optimization. Mathematical Programming 6:62–88
Holmberg K (1990) On the Convergence of Cross Decomposition. Mathematical Programming 47:269–296
Holmberg K (1997a) Linear Mean Value Cross Decomposition: A Generalization of the Kornai-Liptak Method. European Journal of Operational Research 62:55–73
Holmberg K (1997b) Mean Value Cross Decomposition Applied to Integer Programming Problems. European Journal of Operational Research 97:124–138
Karuppiah R, Grossmann IE (2008) A Lagrangean based Branch-and-cut Algorithm for Global Optimization of Nonconvex Mixed-integer Nonlinear Programs with Decomposable Structures. Journal of Global Optimization 41:163–186
Kelley JE (1960) The Cutting-Plane Method for Solving Convex Programs. Journal of the Society for Industrial and Applied Mathematics 8:703–712
Kiwiel KC (1990) Proximity Control in Bundle Methods for Convex Nondifferentiable Minimization. Mathematical Programming 46:105–122
Lemarechal C (1974) An Algorithm for Minimizing Convex Functions. In: Proceedings IFIP'74 Congress pp 552–556
Lima R, Grossmann IE (2011) Computational advances in solving Mixed Integer Linear Programming problems. AIDAC pp 151–160
Linderoth J, Shapiro A, Wright S (2006) The Empirical Behavior of Sampling Methods for Stochastic Programming. Annals of Operations Research 142:215–241

Magnanti TL, Wong RT (1981) Accelerating Benders Decomposition: Algorithmic Enhancement and Model Selection Criteria. Operations Research 29:464–484

Mouret S, Grossmann IE, Pestiaux P (2011) A new Lagrangian Decomposition Approach Applied to the Integration of Refinery Planning and Crude-oil Scheduling. Computers and Chemical Engineering 35:2750–2766

Oliveira F, Gupta V, Hamacher S, Grossmann IE (2013) A Lagrangean Decomposition Approach for Oil Supply Chain Investment Planning under Uncertainty with Risk Considerations. Computers & Chemical Engineering 50:184–195

Saharidis GKD, Minoux M, Ierapetritou MG (2010) Accelerating Benders Method Using Covering Cut Bundle Generation. International Transactions in Operational Research 17:221–237

Sahinidis NV, Grossmann IE (1991) Convergence Properties of Generalized Benders Decomposition. Computers & Chemical Engineering 15:481–491

Snyder LV, Daskin MS (2005) Reliability Models for Facility Location: The Expected Failure Cost Case. Transportation Science 39(3):400–416

Sohn HS, Bricker DL, Tseng TL (2011) Mean Value Cross Decomposition for Two-Stage Stochastic Linear Programming with Recourse. The Open Operational Research Journal 5:30–38

Tarhan B, Gupta V, Grossmann IE (2013) Improving Dual Bound for Stochastic MILP Models Using Sensitivity Analysis. submitted to European Journal of Operational Research

Van Roy TJ (1983) Cross Decomposition for Mixed Integer Programming. Mathematical Programming 25:46–63

Van Slyke R, Wets RJB (1969) L-shaped Linear Programs with Applications to Optimal Control and Stochastic Programming. SIAM Journal on Applied Mathematics 17:638–663

Zowe J (1985) Nondifferentiable Optimization. In K Schittkowski, Computational Mathematical Program, NATO ASI Series F: Computer and Systems Science 15:323–356

A Appendix: Proofs

A.1 Derivation of (36) as valid inequality for (BMP)

Proposition 1 *Equation (36) is a valid inequality that does not cut off the optimal solution of (SPNAC) from (BMP).*

Proof Let $z^{*k}_{LD,s} = \tau_s(c^T\tilde{x}^k_s + d^T\tilde{y}^k_s) + \mu^k H_s \tilde{x}^k_s$ be the solution of the Lagrangean dual for scenario s (LD^k_s), for a given Lagrangean multiplier μ^k, $k \in K$. We can then formulate the following inequality using the definition of the objective function of (LD^k_s):

$$z^{*k}_{LD,s} \leq \tau_s(c^T x_s + d^T y_s) + \mu^k H_s x_s \qquad \forall s \in S,\ k \in K \tag{60}$$

Since (LD^k_s) is a relaxation for scenario s of (SPNAC), the following inequality holds (a proof can be found in Karuppiah and Grossmann (2008), where x_s is replaced by the original x, which is defined as per (SPNAC).

$$z^{*k}_{LD,s} \leq \tau_s(c^T x + d^T y_s) + \mu^k H_s x \qquad \forall s \in S,\ k \in K \tag{61}$$

The inequality (61) obviously also holds for the minimization over y_s:

$$z^{*k}_{LD,s} \leq \min_{y_s} \left\{ \tau_s (c^T x + d^T y_s) + \mu^k H_s x \right\} \qquad \forall s \in S, \; k \in K \tag{62}$$

which can be reformulated using strong duality:

$$z^{*k}_{LD,s} \leq \max_{u_s, v_s} \left\{ \tau_s c^T x + (A_1 x - b_1)^T u_s - b_s^T v_s + \mu^k H_s x \right\} \qquad \forall s \in S, \; k \in K \tag{63}$$

With $(u_s^{k'}, v_s^{k'})$ as the optimal solution in the k'^{th} Benders subproblem, we can rewrite (63) as:

$$z^{*k}_{LD,s} \leq \tau_s c^T x + (A_1 x - b_1)^T u_s^{k'} - b_s^T v_s^{k'} + \mu^k H_s x \qquad \forall s \in S, \; k \in K, \; k' \in K \tag{64}$$

which can be reformulated in the following way by applying (35):

$$z^{*k}_{LD,s} \leq \theta_s + \mu^k H_s x \qquad \forall s \in S, \; k \in K \tag{65}$$

which proves that (36) is a valid inequality. □

A.2 Derivation of (51) as valid inequality for (LMP)

Proposition 2 *Equation (51) is a valid inequality for the Lagrangean dual master problem (LMP).*

Proof Using Minkowski's Theorem, we can express the primal variables $(x,y)^T$ in the following way using the disaggregated expression in terms of scenarios s:

$$(x,y)^T = \sum_{s \in S} \sum_{k \in \bar{K}} \bar{\alpha}_s^k (\bar{x}_s^k, \bar{y}_s^k)^T \tag{66}$$

with $\sum_{k \in \bar{K}} \bar{\alpha}_s^k = 1, \; \forall s \in S, \; 0 \leq \bar{\alpha}_s^k \leq 1$, where $(\bar{x}_s^k, \bar{y}_s^k)$ are all the extreme points of the convex hull of the set of non-complicating constraints Z_s of (SPNAC), which is assumed to be compact, and include all but the non-anticipativity constraints:

$$Z_s = \{A_0 x_s \leq b_0, \; A_1 x_s + B_1 y_s \leq b_1, \; B_s y_s \leq b_s, \; x_s \in X, \; y \geq 0\} \tag{67}$$

While not all of the extreme points are known, we can consider a subset $K \subset \bar{K}$, which can be constructed according to previous solutions from the Lagrangean dual $(\tilde{x}_s^k, \tilde{y}_s^k)$ (Frangioni, 2005). The set of solutions from the Lagrangean dual is augmented with $|K|$ previous solutions from the Benders primal problem $(\hat{x}^k, \hat{y}_s^k)$, which are also feasible in Z_s but are not necessarily extreme points of Z_s. Therefore, (66) can be rewritten as:

$$(x,y)^T = \sum_{s \in S} \left(\sum_{k \in K} \tilde{\alpha}_s^k (\tilde{x}_s^k, \tilde{y}_s^k)^T + \sum_{k \in K} \hat{\alpha}_s^k (\hat{x}^k, \hat{y}_s^k)^T \right) \tag{68}$$

with $\sum_{k \in K} \tilde{\alpha}_s^k + \sum_{k \in K} \hat{\alpha}_s^k = 1$. With this, we can formulate a restricted version of the Dantzig-Wolfe primal master problem (Guignard, 2003; Frangioni, 2005) of (SPNAC):

$$\min_{\alpha} \; \sum_{s \in S} \left(\sum_{k \in K} \tilde{\alpha}_s^k \tau_s (c^T \tilde{x}_s^k + d^T \tilde{y}_s^k) + \sum_{k \in K} \hat{\alpha}_s^k \tau_s (c^T \hat{x}^k + d^T \hat{y}_s^k) \right) \tag{69}$$

$$\text{s.t.} \; -\sum_{s \in S} \left(\sum_{k \in K} \tilde{\alpha}_s^k (H_s \tilde{x}_s^k) + \sum_{k \in K} \hat{\alpha}_s^k (H_s \hat{x}^k) \right) = 0 \qquad (\mu) \tag{70}$$

$$\sum_{k \in K} \tilde{\alpha}_s^k + \sum_{k \in K} \hat{\alpha}_s^k = 1 \qquad (\kappa_s) \qquad \forall s \in S \tag{71}$$

$$\tilde{\alpha}_s^k, \;\; \hat{\alpha}_s^k \geq 0 \qquad \forall s \in S, \; k \in K \tag{72}$$

Note that we can replace the objective (69) in the following way, using the previously obtained objective function values:

$$\min_{\alpha} \sum_{s \in S} \left(\sum_{k \in K} \tilde{\alpha}_s^k z_{LD,s}^{*k} + \sum_{k \in K} \hat{\alpha}_s^k z_{P,s}^{*k} \right) \quad (73)$$

The restricted Dantzig-Wolfe primal master problem yields an upper bound on the Dantzig-Wolfe primal master problem. With the dual variables (μ, κ_s), which are both unrestricted, the Lagrangean dual master problem can be written as:

$$\max_{\kappa,\mu} \quad \sum_{s \in S} \kappa_s \quad (74)$$

$$\text{s.t.} \quad \kappa_s \leq z_{LD,s}^{*k} + \mu H_s \tilde{x}_s^k \qquad \forall s \in S, \; k \in K \quad (75)$$

$$\kappa_s \leq z_{P,s}^{*k} + \mu H_s \hat{x}^k \qquad \forall s \in S, \; k \in K \quad (76)$$

from which we can extract equation (76) as valid inequalities since we derived the Lagrangean master problem (LMP) in (74)-(76) (without quadratic trust-region stabilization term). □

A.3 Proof of convergence for the enhanced cross-decomposition with primal dual multi-cuts

Proposition 3 *The algorithm converges to the optimal solution of (SPNAC) within ε-tolerance in a finite number of steps.*

Proof The algorithm relies on Benders decomposition with multi-cuts to obtain lower and upper bounds. The Benders primal master problem is modified by adding equation (36). However, by proposition 1, equation (36) is a valid inequality that does not cut off the optimal solution. Therefore, we will find the optimal solution within ε-tolerance in a finite number of steps according to the convergence proof by Birge and Louveaux (1988). □

B Appendix: Model for illustrative example

B.1 Problem Description

We applied the cross-decomposition scheme to a facility location problem with unreliable facilities. The reliable facility location problem (RFLP) was introduced by Snyder and Daskin (2005). Their formulation minimizes the sum of investment cost and expected transportation costs in a network subject to facility disruptions. Besides the DC candidate locations, a fictitious facility that is always available is included to model penalties for unsatisfied demands. The problem assumes that all facilities have the same associated probability (q) of being disrupted.

The (RFLP) formulation has been adapted by Garcia-Herreros et al (2014) to include the capacity design of facilities. In this setting, investment costs have two components: fixed-charges and variable charges proportional to the capacity. This formulation also accommodates location-dependent failure probabilities in the scenario generation.

The model is intended to anticipate demand assignments over a discrete set of scenarios. The scenarios represent the possible combinations of active and disrupted locations; their probabilities are parameters calculated from the disruption probabilities of individual DCs by assuming independence among locations. Under the assumption of equal disruption probability (q) at the candidate locations, the probability of a scenarios only depends on the number of disrupted locations. For a problem with N candidate locations, the probability (τ_s) of one scenario with n simultaneous disruptions can be calculated according to equation (77).

$$\tau_s = q^n (1-q)^{N-n} \quad (77)$$

The following notation is used in the two-stage stochastic programming formulation. The set of candidate locations for DCs is denoted by J, including a fictitious DC with unlimited capacity in position $|J|$

(note that $|J| = N+1$); the set of demand sites is denoted by I; the set of scenarios is denoted by S. The decision whether DC at candidate location j is selected is represented by the binary variable x_j; the fraction of demand i satisfied from location j in scenario s is denoted by $y_{s,j,i}$; the capacity of DC at location j is denoted by c_j. The parameters of the problem are: the fixed-charge for the selection of DCs j (F_j), the unit capacity cost for DC j (V_j), the demand at site i (D_i), the unit transportation cost from DC j to site i $(A_{j,i})$, the probability of scenario s (τ_s), the maximum capacity of DCs (C^{max}), and the matrix indicating the availability of DC j in scenario s $(T_{s,j})$.

B.2 Model

According to Garcia-Herreros et al (2014), the capacitated reliable facility location problem (CRFLP) is formulated as follows.

$$\textit{(CRFLP)} \quad \min \sum_{j}^{|J|-1} (F_j x_j + V_j c_j) + \sum_{s\in S}\sum_{j\in J}\sum_{i\in I} \tau_s A_{j,i} D_i y_{s,j,i} \tag{78}$$

$$\text{s.t. } c_j - C^{max} x_j \leq 0 \qquad \forall j < |J| \tag{79}$$

$$\sum_{i\in I} D_i y_{s,j,i} - T_{s,j} c_j \leq 0 \qquad \forall s \in S,\ \ j \in J \tag{80}$$

$$\sum_{j\in J} y_{s,j,i} = 1 \qquad \forall s \in S,\ \ i \in I \tag{81}$$

$$x_j \in \{0,1\},\ \ 0 \leq c_j \leq C^{max} \qquad \forall j \in J \tag{82}$$

$$0 \leq y_{s,j,i} \leq 1 \qquad \forall s \in S,\ \ j \in J,\ \ i \in I \tag{83}$$

The objective function (78) minimizes the sum of investment cost and expected transportation cost. Constraints (79) ensure that capacity is only allocated to selected DCs. Constraints (80) limit demand satisfaction to the inventory availability according to the binary parameter $T_{s,j}$ that indicates the disrupted locations in each scenario. Constraints (81) enforce demands satisfaction or penalization in every scenario. The domain of the variables is presented in constraints (82) and (83).

The formulation (CRFLP) is know to have a poor linear relaxation. In order to strengthen the MILP formulation, a redundant set of constraints can be added to the model. The set of tightening constraints (86) directly prevent demand assignments to DCs that are not selected or disrupted.

$$\textit{(CRFLP-t)} \quad \min \sum_{j}^{|J|-1} (F_j x_j + V_j c_j) + \sum_{s\in S}\sum_{j\in J}\sum_{i\in I} \tau_s A_{j,i} D_i y_{s,j,i} \tag{84}$$

$$\text{s.t. } (79) - (83) \tag{85}$$

$$y_{s,j,i} - T_{s,j} x_j \leq 0 \qquad \forall s \in S,\ \ j \in J,\ \ i \in I \tag{86}$$

B.3 Data

The data for the case study is taken from Daskin (1995). The original problem considers 49 US cities that simultaneously serve as demand sites and potential distribution centers (DCs). Their demand for a single commodity is assumed to be proportional to the state populations in 1990. The original formulation includes uncapacitated DCs with investment costs estimated from the real-state market. Variable costs associated with the DC capacities have been added at a rate of \$0.0001 per unit of product. The transportation costs are proportional to the great-circle distance between facilities.

Given the very large number of possible scenarios (2^{49}), we have selected subsets of 10, 11, and 12 facilities as candidate locations for DCs in the different instances of the problem. The locations included in the smallest instance are: Sacramento (CA), Albany (NY), Austin (TX), Tallahassee (FL), Harrisburg (PA), Springfield (IL), Columbus (OH), Montgomery (AL), Salem (OR), and Des Moines (IA). The additional location included in the instance with 11 candidate DCs is Lansing (MI). The largest instance with 12 candidate DCs also includes Trenton (NJ).

Given the very small probability of scenarios with more than 5 simultaneous disruptions, they have been grouped into a single scenario in which all demands are penalized. The effect of this approximation is limited by the magnitude of corresponding probabilities. Furthermore, the approximation improves the numerical performance of the algorithm. Despite the reduction in size, the problem still implies minimizing the cost over large sets of scenarios. The failure probability of all DCs has been left to the value originally used by Snyder and Daskin (2005): $q = 0.05$.