A cross-decomposition scheme with integrated primal–dual multi-cuts for two-stage stochastic programming investment planning problems

Sumit Mitra\textsuperscript{1} · Pablo Garcia-Herreros\textsuperscript{1} · Ignacio E. Grossmann\textsuperscript{1}

Received: 9 February 2014 / Accepted: 5 March 2016 / Published online: 2 April 2016
© Springer-Verlag Berlin Heidelberg and Mathematical Optimization Society 2016

Abstract We describe a cross-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 a novel 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 numerical experiments on a number of instances of a facility location problem under disruptions. In the original formulation, where the underlying LP relaxation is weak, the cross-decomposition method outperforms multi-cut Benders decomposition. If the formulation is improved with the addition of tightening constraints, the performance of both decomposition methods improves but cross-decomposition clearly remains the best method for large-scale problems.

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

Electronic supplementary material The online version of this article (doi:10.1007/s10107-016-1001-y) contains supplementary material, which is available to authorized users.

\textsuperscript{1} Ignacio E. Grossmann
grossmann@cmu.edu

\textsuperscript{1} Department of Chemical Engineering, Carnegie Mellon University, 5000 Forbes Ave., Pittsburgh, PA 15213, USA
1 Motivation

Two-stage stochastic programming investment planning problems [5] can be hard to solve since the resulting deterministic equivalent programs can lead to very large-scale problems. There are two main approaches, which are complementary techniques to address the computational challenge: First, sampling methods [34] and scenario reduction techniques [22] can be used to limit the number of scenarios. Second, decomposition schemes, such as Benders decomposition also known as L-shaped method [3,17,44] or Lagrangean decomposition [9,20] can be applied to exploit decomposable problem structure. In the following we focus on decomposition schemes 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 [35,39]. 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 [4], Pareto optimal cuts [35] and cut bundle generation methods [38] (among others) have been proposed.

Lagrangean decomposition applied to stochastic programming problems is a special form of Lagrangean relaxation, which relaxes the so-called non-anticipativity constraints that enforce the same first-stage decisions across scenarios [9]. 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 [13,23,24] or cutting planes [11,30] can be a bottleneck that slows down the overall convergence of the algorithm. Techniques to speed up convergence of the multiplier update include the bundle method [31,32,46], the volume algorithm [2] and the analytic center cutting plane method [18]. There are also strategies to combine bounds from subgradients with cutting planes [36,37], and to update the multipliers based on dual sensitivity analysis [21,42].

However, while there are many papers that focus on improvements for one of the two decomposition schemes, it seems that only few research efforts address combining their complementary strengths. First proposed by Van Roy [43], 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 [25] 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 seen as a hard task. Some variations of the method, e.g. mean value cross-decomposition [26,28] 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 for stochastic programming problems that uses cross-decomposition.\textsuperscript{1} On the one hand, Cerisola et al. [10] 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. [41] derive a mean value cross-decomposition approach for two-stage stochastic programming problems (LP) based on Holmberg [27], 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.\textsuperscript{2} 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 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 and 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 a novel 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 decisions are continuous. Once first-stage decisions are fixed, the problem decomposes into scenarios (Benders primal subproblems). The Lagrangean dual subproblem for each scenario is obtained by dualizing non-anticipativity constraints. Both, the primal and the dual subproblem

\textsuperscript{1} Interestingly, only 49 results are found by a Google Scholar search on “cross-decomposition” + “stochastic programming” as of January 18th 2014. Thereof, only two publications seem to actively combine Benders and Lagrangean decomposition in a somewhat similar way to our proposed scheme.

\textsuperscript{2} Although, the importance of a good formulation should not be neglected. See Bixby and Rothberg [6] and Lima and Grossmann [33] for further details and examples.
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 cuts obtained in previously solved Benders primal subproblems. The main motivation in both cases is to strengthen the bounds predicted by both master problems.

2 Problem statement

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

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

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 \(\tau_s\) is the probability for scenario \(s \in S\) with \(\sum_{s \in S} \tau_s = 1\). The first-stage decisions, \(x\), which are of dimension \(m\) are mixed-integer and correspond to discrete choices for investments and associated capacities:

\[
X = \left\{ x = \left(\begin{array}{c} x_1 \\ x_2 \end{array}\right) : x_1 \in \{0, 1\}^n, \; x_2 \geq 0 \right\} \quad (7)
\]

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

Note that the problem naturally decomposes into scenarios once the investment decisions \(x\) are fixed. 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 [9] and re-write problem (SP) in the following way \((\text{SPNAC})\), where Eq. (12) represents the non-anticipativity constraints, \((x_1 = x_2 = \cdots = x_{|S|})\), with suitable matrices \((H_1, \ldots, H_{|S|})\) where each \(H_s\) has dimension \((|S| - 1)m \times m\):
A cross-decomposition for two-stage stochastic programming

\[ \min \quad TC = \sum_{s \in S} \tau_s (c^T x_s + d^T y_s) \] (8)

s.t. \[ A_0 x_s \leq b_0 \] (9)
\[ A_1 x_s + B_1 y_s \leq b_1 \quad \forall s \in S \] (10)
\[ B_s y_s \leq b_s \quad \forall s \in S \] (11)
\[ \sum_{s \in S} H_s x_s = 0 \] (12)
\[ x_s \in X_s \quad \forall s \in S \] (13)
\[ y_s \geq 0 \quad \forall s \in S \] (14)

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

\[ X_s = \left\{ x_s = \left( x_{1,s}, x_{2,s} \right) : x_{1,s} \in \{0, 1\}^n, \ x_{2,s} \geq 0 \right\} \quad \forall s \in S \] (15)

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

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

\[ \square \]

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, while being strengthened with additional cuts that are generated from each other. 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 \((BSP^p_s)^k\) and a dual formulation \((BSP^d_s)^k\), which are equivalent since \((BSP^p_s)^k\) is an LP that can be solved with zero duality gap. We present both in
the following.

\[
(BSP p^k_s) \quad \text{min} \quad \tau_s (c^T \hat{x}^k + d^T_s y_s) \\
\text{s.t.} \quad B_1 y_s \leq b_1 - A_1 \hat{x}^k \\
B_s y_s \leq b_s \\
y_s \geq 0
\] (16)

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 \((BSP p^k_s)\) can be written as:

\[
(BSP d^k_s) \quad \text{max} \quad \tau_s c^T \hat{x}^k + (A_1 \hat{x}^k - b_1)^T u_s - b^T_s v_s \\
\text{s.t.} \quad -u_s^T B_1 - v_s^T B_s \leq \tau_s d_s \\
u_s, v_s \geq 0
\] (20)

Let \( \tilde{x}^k_s, \tilde{y}^k_s \) be the optimal solution of \((BSP p^k_s)\) and \( (u^k_s, v^k_s) \) be the solution of \((BSP d^k_s)\). Let us define \( z^*_{SP, s} \), in the following way (equivalence due to strong duality):

\[
z^*_{P, s} = \tau_s (c^T \hat{x}^k + d^T_s \tilde{y}^k_s) = \tau_s c^T \hat{x}^k + (A_1 \hat{x}^k - b_1)^T u^k_s - b^T_s v^k_s \quad \forall s \in S, k \in K
\] (23)

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

\[
TC \leq z^*_{P} \quad \forall k \in K
\] (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 [9, 20]. The Lagrangean Dual \((LD^k_s)\) is formulated in the following with given fixed Lagrange multipliers \( \mu^k \) for convenience defined as row vectors each scenario \( s \), denoted as \((LD^k_s)\).

\[
(LD^k_s) \quad \text{min} \quad \tau_s (c^T x_s + d^T_s y_s) + \mu^k H_s x_s \\
\text{s.t.} \quad A_0 x_s \leq b_0 \\
A_1 x_s + B_1 y_s \leq b_1 \\
B_s y_s \leq b_s \\
x_s \in X \\
y_s \geq 0
\] (25)

Let \( \tilde{x}^k_s, \tilde{y}^k_s \) be the optimal solution of \((LD^k_s)\). Let us define \( z^*_{LD, s} \) in the following way:

\[
z^*_{LD, s} = \tau_s (c^T \tilde{x}^k_s + d^T_s \tilde{y}^k_s) + \mu^k H_s \tilde{x}^k_s \quad \forall s \in S, k \in K
\] (31)
A cross-decomposition for two-stage stochastic programming… 101

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

$$z^*_{LD} \leq TC \quad \forall k \in K \quad (32)$$

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 $\hat{x}^{k+1}$, 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 $(BSPd^k_s)$, 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 $(BSPd^k_s)$ (across all scenarios $s$) is a lower bound on the optimal value of the primal $(BSPp^k_s)$ 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 [3]

$$TC \geq \sum_{s \in S} \tau_s c^T x + \sum_{s \in S} (A^1 x - b^1)^T u^k_s - \sum_{s \in S} b^T_s v^k_s \quad \forall k \in K \quad (33)$$

The second cut, presented in Eqs. (34)–(35) is the Benders multi-cut [4], where $\theta_s$ is the objective function value for scenario $s$.

$$TC \geq \sum_{s \in S} \theta_s \quad (34)$$

$$\theta_s \geq \tau_s c^T x + (A^1 x - b^1)^T u^k_s - b^T_s v^k_s \quad \forall s \in S, \ k \in K \quad (35)$$

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) (see [45]). 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 $\theta_s$ based on the previously obtained information.
from the Lagrangean dual:
\[ z^*_{LD,s} \leq \theta_s + \mu^k H_s x \quad \forall s \in S, \; k \in K. \]  

(36)

The proof that (36) is a valid inequality for the Benders primal master problem can be found in Appendix “Derivation of (36) as valid inequality for (BMP)”

**No-Good Cut** If the complicating variables, \( x \), are all binary variables, we can also formulate a no-good cut [1] to exclude previously visited solutions from the solution space:

\[ \sum_{l \in L^k_0} x_l - \sum_{l \in L^k_1} x_l - |L^k_1| + 1 \leq 0 \quad \forall k \in K \]  

(37)

In Eq. (37), \( L^k_0 \) and \( L^k_1 \) are defined as follows (where \( x_l \) is the \( l \)th component of \( x \)):

\[ L^k_0 = \{ l : x_l = 0 \text{ in iteration } k \} \]  

(38)

\[ L^k_1 = \{ l : x_l = 1 \text{ in iteration } k \} \]  

(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 \((BMP_k^{k+1})\) that yields the next primal vector \( \hat{x}^{k+1} \) as follows:

\[
(BMP^{k+1}) \min \eta_{BMP} \\
\text{s.t. } \eta_{BMP} \geq \sum_{s \in S} \theta_s \\
\theta_s \geq \tau^k_s c^T x + (A_1 x - b_1)^T u^k_s - b^T u^k_s \quad \forall s \in S, \ k \in K \]  

(40) \hspace{1cm} (41)

\[ \theta_s \geq z^*_{LD,s} - \mu^k H_s x \quad \forall s \in S, \ k \in K \]  

(42)

\[ A_0 x \leq b_0 \]  

(43)

\[ x \in X, \; \eta_{BMP} \in \mathbb{R}^1, \; \theta_s \in \mathbb{R}^1 \quad \forall s \in S \]  

(44)

(45)

If \( X \) has only integer variables, we can add the no-good cut (37) with the sets \( L^k_0 \) and \( L^k_1 \) defined by (38)–(39) to \((BMP^{k+1})\). Let \( z^*_{BMP}^{k+1} \) 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):

\[ z^*_{BMP}^{k+1} \leq TC \quad \forall k \in K \]  

(46)

Note that by adding Eq. (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

\[ z^{*k'}_{LD} = \sum_{s \in S} z^{*k'}_{LD,s} - \mu^{k'} \sum_{s \in S} H_s x_s \leq \sum_{s \in S} \theta_s \leq z^{*k+1}_{BMP} \quad \forall \ k \in K, \ k' \in K, \ k' \leq k. \]  

(47)

3.2.2 Lagrangean (dual) master problem (multiplier update)

Once the Lagrangean dual subproblems are solved, a new solution for the Lagrangean multipliers \( \mu \) 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 [13,23,24]. Unfortunately, the convergence of the subgradient method is not very reliable, especially when dealing with large-scale problems. One alternative to the subgradient algorithm is using 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 [15,19].

Typical cutting plane The typical form of the cutting plane [11,30] is as follows:

\[ TC \leq \sum_{s \in S} \tau_s (c^T \tilde{x}_s^k + d_s^T \tilde{y}_s^k) + \mu H_s \tilde{x}_s^k \quad \forall k \in K, \]  

(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 Eq. (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 [15,19], in which \(\kappa_s\) is the variable associated with the objective function value in scenario \(s\):  

\[ TC \leq \sum_{s \in S} \kappa_s \]  

(49)

\[ \kappa_s \leq \tau_s (c^T \tilde{x}_s^k + d_s^T \tilde{y}_s^k) + \mu H_s \tilde{x}_s^k \quad \forall s \in S, \ k \in K \]  

(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 [15]. 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 $\kappa_s$ of each scenario $s$ in the Lagrangean master problem since we will use the disaggregated version of the Lagrangean master problem:

$$\kappa_s \leq z_{P,s}^k + \mu H_s \hat{x}_s^k \quad \forall s \in S, \ k \in K$$  \hspace{1cm} (51)

The proof that (51) is a valid inequality for the Lagrangean dual master problem is given in Appendix “Derivation of (51) as valid inequality for (LMP)”

**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 ($LM P^{k+1}$), which yields the next Lagrangean multipliers $\mu^{k+1}$, can be formulated as:

$$(LM P^{k+1}) \max \eta_{LMP} + \frac{\delta}{2} \| \mu - \bar{\mu} \|^2 \quad \text{s.t.} \quad \eta_{LMP} \leq \sum_{s \in S} \kappa_s$$

$$\kappa_s \leq \tau_s (c^T \hat{x}_s^k + d^T_s \hat{y}_s^k) + \mu H_s \hat{x}_s^k \quad \forall s \in S, \ k \in K$$

$$\kappa_s \leq z_{P,s}^k + \mu H_s \hat{x}_s^k \quad \forall s \in S, \ k \in K$$

$$\eta_{LMP} \in \mathbb{R}^1, \ \mu \in \mathbb{R}^{1 \times (|S| - 1)m}, \ \kappa_s \in \mathbb{R}^1 \quad \forall s \in S$$

where $|S|$ is the number of scenarios and $m$ the number of dualized first-stage variables. The objective function (52) contains the additional quadratic stabilization term $\frac{\delta}{2} \| \mu - \bar{\mu} \|^2$ that defines a trust-region for the update of the Lagrangean multipliers [14,31,32,46]. The stabilization requires initial values and update strategies for the penalty $\delta$, 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 ($LM P^{k+1}$) without the stabilization term. 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:
\[ z^{*k+1}_{LMP} \leq \sum_{s \in S} \kappa_s \leq \sum_{s \in S} z^{*k}_{p,s} + \mu \sum_{s \in S} H_s \hat{x}^k = z^*_{p,k'} \quad \forall k \in K, \ k' \in K, \ k' \leq k. \]  

(57)

### 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:

\[ \hat{x}^{k=0} = 0 \quad \text{(58)} \]

\[ \mu^{k=0} = 0 \quad \text{(59)} \]

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 the problem has complete recourse. Furthermore, initial values for the trust-region penalty \( \delta \) 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**
   
   \[ k = 0, \ K = \{0\}, \ \hat{x}^{k=0} = 0, \ \mu^{k=0} = 0, \ LB = -\infty, \ UB = \infty, \ \text{set } \varepsilon \geq 0, \]
   
   Go to step 2

2. **Benders primal subproblem**
   
   For given \( \hat{x}^k \), solve \( (BSP_{p,k}') \), \( \forall s \in S \) in parallel
   
   Store \( (u^k, v^k) \), and generate sets \( L^k_0, L^k_1 \) if \( X \) has only binary variables
   
   If \( z_{p,k}^* < UB \), set \( UB = z_{p,k}^* \)

3. **Lagrangean dual subproblem**
   
   For given \( \mu^k \), solve \( (LD_{s,k}') \), \( \forall s \in S \) in parallel
   
   Store solution \( \hat{x}_{s,k}, \hat{y}_{s,k} \) and \( z_{LD,s,k}^* \)
   
   Go to step 4

4. **Lagrangean (dual) master problem**
   
   For given \( \hat{x}_{s,k}, \hat{y}_{s,k}, z_{p,s,k}^*, \forall s \in S, \ k \in K \), solve \( (LMP^{k+1}_s) \)
   
   Store solution for Lagrangean multipliers \( \mu \) and set them as \( \mu^{k+1} \)
   
   Update \( \delta \) and \( \bar{\mu} \) according to update strategies
   
   Go to step 5

5. **Benders (primal) master problem**
   
   For given \( (u^k, v^k), (L^k_0, L^k_1), z_{LD,s,k}^* \), \( \forall s \in S, \ k \in K \), solve \( (BMP^{k+1}_s) \)
Fig. 1 Algorithmic flow (bold arrows) and flow of information (thin arrows) for our cross-decomposition scheme with multi-cuts

Store solution for first-stage primal investment decisions \( \hat{x} \) and set them as \( \hat{x}^{k+1} \)
Set \( LB = z_{BMP}^{*k+1} \)
Go to step 6

6. **Optimality check**
   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 [25,43] 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. In contrast, our algorithm solves both master problems at each iteration rather than avoiding them. Note that the convergence tests used in the original cross-decomposition scheme [25,43] are no longer needed; convergence towards optimality is guaranteed as proved in Appendix “Proof of convergence for the enhanced cross-decomposition with primal dual multi-cuts”. Additionally, Van Roy [43] and Holmberg [25] do not consider the case of stochastic programming and its decomposable structure as well as associated difficulties in the direct exchange of primal and dual variables between subproblems, which can for instance also be found in Lagrangean Decomposition. Furthermore, in the original cross-decomposition algorithm no strengthening techniques such as multi-cuts or valid inequalities [i.e. valid inequalities (36) and (57)] are introduced to the respective master problems (since they are regarded as hard to solve)—the latter ones are a novel contribution of this paper.

6 Facility location with disruptions

In order to test the proposed algorithm, we apply the cross-decomposition scheme to a number of instances of 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 capacitated reliable facility location problem (CRFLP) is formulated as a two-stage stochastic program. First-stage decisions involve the selection of DCs and their capacities. Second-stage decisions are the demand assignments in scenarios that establish the combination 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 2”: CRFLP and CRFLP-t. The second model (CRFLP-t) includes a redundant set of constraints that tightens 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 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 [4] and our cross-decomposition algorithm, as previously described in Sect. 4, in GAMS 24.3.1 [7].
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 [8] on the 12 processors of an Intel Xeon (2.67 GHz) machine with 16 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 5.6.3 to use the processors as parallel threads. For the fullspace implementations, 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 5.6.3. We set quad 1 (for quad precision) and numericfocus 3 (for high attention to numerical issues) in all master problems and subproblems. The optimality tolerances for reduced costs are set to $10^{-7}$. We use barhomogeneous 1 to detect infeasibilities with the barrier method employed to solve the QP Lagrangean master problems. Instances are considered solved when their relative optimality gap is below $10^{-4}$; the MILPs inside the Benders and cross-decomposition loops are solved using an optimality tolerance of $10^{-6}$. We set a wall-clock time limit of 10,000 s for the solution of each instance.

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 $\delta$ (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 $\delta$ is updated according to 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 [31] and Frangioni [14], 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 decomposition, and cross-decomposition) with 10 random instances generated for three cases with different number of candidate DCs and scenarios. Both problem formulations (CRFLP and CRFLP-t) presented in “Appendix 2” are tested in order to analyze the impact of the LP relaxation in the performance of the solution methods. The instances are generated based on the data set presented by Snyder and Daskin [40] by sampling transportation costs and demands from uniform distributions bounded between 80 and 120% of the original values. In particular, we use random transportation costs to include different cost structures in the objective function and random demands to analyze different shapes of the feasible region; the evaluation of the solution methods on a large number of instances with different values for key parameters allows establishing their variability in performance. The details of the methodology used to generate the random instances can be found in “Appendix 2”.

© Springer
Table 1  Computational statistics for the 10 instances of each fullspace formulation

| Model   | DCs (N) | Scenarios | Constraints | Variables | Binary variables | Mean objective ($) | Mean LP relaxation gap (%) |
|---------|---------|-----------|-------------|-----------|------------------|--------------------|---------------------------|
| CRFLP   | 10      | 639       | 38,992      | 345,084   | 10               | 970,182            | 49.68                     |
| CRFLP-t | 10      | 639       | 383,413     | 345,084   | 10               | 970,182            | 0.55                      |
| CRFLP   | 11      | 1025      | 63,564      | 603,751   | 11               | 978,036            | 51.89                     |
| CRFLP-t | 11      | 1025      | 666,264     | 603,751   | 11               | 978,036            | 0.67                      |
| CRFLP   | 12      | 1587      | 99,996      | 1,012,534 | 12               | 981,636            | 54.09                     |
| CRFLP-t | 12      | 1587      | 1,110,915   | 1,012,534 | 12               | 981,636            | 0.79                      |

6.2.1 Comparison of methods

In Table 1, we report the computational statistics of the fullspace models for the three cases. It can be observed that an increasing number of DCs implies more scenarios, constraints, and variables. The mean of the optimal solutions for the 10 instances together with the mean of their LP relaxation gap is also presented in Table 1. The statistics show a significant increase in the number of constraints for the CRFLP-t formulations as a result of the addition of tightening constraints; these constraints are also responsible for the improvement in the LP relaxation. It is important to note that despite the size of the formulations, the number of discrete variables remains small because there is only one binary variable per candidate DC.

The performance of the solution methods on the instances are presented in Figs. 2, 3, and 4. These performance curves show the percentage of instances that can be solved to optimality within the time limit shown in the horizontal axis. A complete report of the results obtained from the computational experiments can be found in the Supplementary Material.

Figure 2 compares the wall-clock time required by the methods to solve the instances with 639 scenarios. It can be observed that cross-decomposition on the CRFLP-t formulation outperforms all other methods by solving all instances in less than 800 s. The fullspace CRFLP formulation also solves all instances in less than 800 s, but it takes more than 600 s for most of them. Benders decomposition on the CRFLP-t formulation is very efficient for some instances (5 instances require less than 500 s), but it takes a long time to solve others. Benders decomposition on the CRFLP formulation does not solve any instance within the time limit in the case with 639 scenarios.

Similar trends can be observed in Fig. 3 for the performance of the solution methods on the case with 1025 scenarios. In this case, cross-decomposition on the CRFLP-t formulation clearly outperforms all methods by solving all instances in less than 3250 s. Cross-decomposition on the CRFLP formulation also presents a relatively good performance by solving 7 instances in less than 3000 and all instances in less than 5200 s. The fullspace CRFLP formulation solves all instances in less than 5100 s, but it requires over 4000 s for most of them. Poor performance is observed for the fullspace CRFLP-t formulation and Benders decomposition on both CRFLP and CRFLP-t formulations.
For the case with 1587 scenarios, Fig. 4 shows that cross-decomposition is the only solution method capable of solving some instances. In this case, cross-decomposition on the CRFLP formulation solves 3 instances and cross-decomposition on the CRFLP-t formulation solves 8 instances within the time limit. This results are in line with the trends observed in the previous cases: the cross-decomposition method outperforms all other methods in large-scale problems.

The mean solution time over the 10 instances is presented for each method in Table 2. It can be observed that cross-decomposition on the CRFLP-t formulation is on average the fastest method for all cases. It is interesting to note that Benders
A cross-decomposition for two-stage stochastic programming…

Fig. 4  Performance curves for the solution methods in the case with 1587 scenarios

Table 2  Comparison of mean solution times

| Case (scenarios) | Fullspace CRFLP (s) | Benders CRFLP(s) | Cross CRFLP (s) | Fullspace CRFLP-t (s) | Benders CRFLP-t (s) | Cross CRFLP-t (s) |
|------------------|---------------------|------------------|----------------|-----------------------|---------------------|------------------|
| 639              | 626                 | 10,000*          | 809            | 3070                  | 1048                | 472              |
| 1025             | 4413                | 10,000*          | 3200           | 9783                  | 10,000*             | 1,707            |
| 1587             | 10,000*             | 10,000*          | 9947           | 10,000*               | 10,000*             | 7752             |

* Time limit reached

decomposition on the CRFLP-t formulation shows good results for the instances with 639 scenarios but not for the instances with 1025 and 1587 scenarios. There is also a progressive deterioration of the performance of the fullspace CRFLP formulation from the case with 639 scenarios to the case with 1587 scenarios. This performance confirms that cross-decomposition is the best alternative for large-scale problems. The computational experiments also show that both Benders and cross-decomposition are affected by the underlying LP relaxation of the models. However, if we compare both decomposition strategies on the CRFLP and CRFLP-t formulations, it is clear that the performance of the cross-decomposition method is less affected by the weakness of the formulation. Hence, we conclude that cross-decomposition is less sensitive to weak formulations due to the presence of strong cuts that originate from the Lagrangean dual.

The time spent by the decomposition methods is used to solve subproblems and master problems. For both implementations (CRFLP and CRFLP-t) of Benders decomposition, the wall-clock time spent in the Benders master problems is over 90% of the total time. We also conducted a separate analysis to estimate the impact of parallelization for the Benders and Lagrangean subproblems. As a result of parallelization, the time spent in Benders decomposition is significantly reduced: the speed-up for solving
the Benders subproblems of the CRFLP formulation in 12 parallel threads is around 8.5 in the case with 639 scenarios and around 7.0 for the Benders subproblems of the CRFLP-t formulation with 1587 scenarios. A similar situation occurs in the implementation of the cross-decomposition method. Most of the wall-clock time is spent in solving the Benders master problems: on average, around 58% of the time in the case with 639 scenarios and over 75% in the case with 1587 scenarios. The Lagrangean master problems also require a considerable amount of time: approximately 24% of the total time in the case with 639 scenarios and 12% in the case with 1587 scenarios. There is also a significant decrease in the solution time required to solve the MILP Lagrangean subproblems as a result of parallelization: the speed-up is approximately 5.9 for the Lagrangean subproblems of the CRFLP formulation with 639 scenarios and 6.3 for the Lagrangean subproblems of the CRFLP-t formulation with 1587 scenarios. The result suggests that parallelization of the subproblems yields better speed-ups with LP subproblems that can be solved efficiently by individual threads.

7 Conclusion

We have described a cross-decomposition algorithm that combines Benders and 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. Aside from the unique integration of primal–dual multi-cuts in the master problems, the proposed cross-decomposition scheme is also unique compared to previous work given that it has been developed for solving two-stage stochastic programming problems.

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

Appendix 1: Proofs

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

\[
z^{*k}_{LD,s} \leq \tau_s(c^T x_s + d^T y_s) + \mu_k H_s x_s \quad \forall s \in S, \ k \in K
\]  

(60)

Since \((LD^k_s)\) is a relaxation for scenario \( s \) of \((SPNAC)\), the following inequality holds (a proof can be found in [29]), where \( x_s \) is replaced by the original \( x \), which is defined as per \((SPNAC)\).

\[
z^{*k}_{LD,s} \leq \tau_s(c^T x + d^T y_s) + \mu_k H_s x \quad \forall s \in S, \ k \in K
\]  

(61)

The inequality (61) obviously also holds for the minimization over \( y_s \) [subject to (2), (5), (17)–(19)]:

\[
z^{*k}_{LD,s} \leq \min_{y_s} \{ \tau_s(c^T x + d^T y_s) + \mu_k H_s x \} \quad \forall s \in S, \ k \in K
\]  

(62)

which can be reformulated as a maximization over \((u_s, v_s)\) using strong duality [subject to (2), (5), (21)–(22)]:

\[
z^{*k}_{LD,s} \leq \max_{u_s, v_s} \{ \tau_s c^T x + (A_1 x - b_1)^T u_s - b^T s v_s + \mu_k H_s x \} \quad \forall s \in S, \ k \in K
\]  

(63)

Note that (62) and (63) have exactly the same structure as the Benders subproblems \((BSPp^k_s)\) and \((BSPd^k_s)\) respectively for a given \( x \) with the exception of the constant Lagrangean term \( \mu_k H_s x \). Therefore, for a given \( x = x^{k'} \) that coincides with the input to the \( k' \)th Benders subproblem, (63) has the same optimal solution as the \( k' \)th Benders subproblem. With \((u^{k'}_s, v^{k'}_s)\) as the optimal solution in the \( k' \)th Benders subproblem, we can rewrite (63) using weak duality (in case of variation in \( x \), as in the classical Benders cut) as:

\[
z^{*k}_{LD,s} \leq \tau_s c^T x + (A_1 x - b_1)^T u^{k'}_s - b^T s v^{k'}_s + \mu_k H_s x \quad \forall s \in S, \ k \in K
\]  

(64)

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

\[
z^{*k}_{LD,s} \leq \theta_s + \mu_k H_s x \quad \forall s \in S, \ k \in K
\]  

(65)

which proves that (36) is a valid inequality.

\(\square\)

**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 \(s\):

\[
(x, y)^T = \sum_{s \in S} \sum_{k \in \tilde{K}} \tilde{\alpha}_s^k (\tilde{x}_s^k, \tilde{y}_s^k)^T
\]

(66)

with \(\sum_{k \in \tilde{K}} \tilde{\alpha}_s^k = 1, \forall s \in S, 0 \leq \tilde{\alpha}_s^k \leq 1\), where \((\tilde{x}_s^k, \tilde{y}_s^k)\) 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:

\[
Z_s = \{A_0 x_s \leq b_0, A_1 x_s + B_1 y_s \leq b_1, B_s y_s \leq b_s, x_s \in X, y \geq 0\}
\]

(67)

While not all of the extreme points are known, we can consider a subset \(K \subset \tilde{K}\), which can be constructed according to previous solutions from the Lagrangean dual \((\tilde{x}_s^k, \tilde{y}_s^k)\) [15]. The set of solutions from the Lagrangean dual is augmented with \(|K|\) previous solutions from the Benders primal problem \((\hat{x}_s^k, \hat{y}_s^k)\), which are also feasible in \(Z_s\) but are not necessarily extreme points of \(Z_s\). Therefore, (66) can be rewritten as:

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

(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 [15, 19] of (SPNAC):

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

(69)

s.t. \(-\sum_{s \in S} \left( \sum_{k \in K} \tilde{\alpha}_s^k (H_s \tilde{x}_s^k) + \sum_{k \in K} \hat{\alpha}_s^k (H_s \hat{x}_s^k) \right) = 0\) \hspace{1cm} \((\mu)\)

\[
\sum_{k \in K} \tilde{\alpha}_s^k + \sum_{k \in K} \hat{\alpha}_s^k = 1 \hspace{1cm} \forall s \in S \hspace{1cm} \kappa_s \]

(70)

\[
\tilde{\alpha}_s^k, \hat{\alpha}_s^k \geq 0 \hspace{1cm} \forall s \in S, k \in K
\]

(71)

(72)

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

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

(73)

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

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

subject to

\[
\kappa_s \leq z^*_L D_{s} + \mu H_s \hat{x}_s^k \quad \forall s \in S, \ k \in K \quad (75)
\]

\[
\kappa_s \leq z^*_P s + \mu H_s \hat{x}_s^k \quad \forall s \in S, \ k \in K \quad (76)
\]

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

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 \(\varepsilon\)-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 Eq. (36). However, by proposition 1, Eq. (36) is a valid inequality that does not cut off the optimal solution. Therefore, we will find the optimal solution within \(\varepsilon\)-tolerance in a finite number of steps according to the convergence proof by Birge and Louveaux [4].

Appendix 2: Model for facility location with disruptions

**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 [40]. 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. [16] 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 \( (\tau_s) \) of one scenario with \( n \) simultaneous disruptions can be calculated according to Eq. (77).

\[
\tau_s = q^n(1 - q)^{N-n}
\]  

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 \( i \) is 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 \( s \) (\( \tau_s \)), the maximum capacity of DCs (\( C_{max} \)), and the matrix indicating the availability of DC \( j \) in scenario \( s \) (\( T_{s,j} \)).

**Model**

According to Garcia-Herreros et al. [16], the capacitated reliable facility location problem (CRFLP) is formulated as follows.

\[
\begin{align*}
\text{(CRFLP)} \quad \min & \quad \sum_{j}(|J|-1)F_j x_j + \sum_{s \in S} \sum_{j \in J} \sum_{i \in I} \tau_s A_{j,i} D_i y_{s,j,i} \\
\text{subject to} & \quad c_j - C_{max} x_j \leq 0 & \forall j < |J| \quad (79) \\
& \quad \sum_{i \in I} D_i y_{s,j,i} - T_{s,j} c_j \leq 0 & \forall s \in S, \ j \in J \quad (80) \\
& \quad \sum_{j \in J} y_{s,j,i} = 1 & \forall s \in S, \ i \in I \quad (81) \\
& \quad x_j \in \{0, 1\}, \ 0 \leq c_j \leq C_{max} & \forall j \in J \quad (82) \\
& \quad 0 \leq y_{s,j,i} \leq 1 & \forall s \in S, \ j \in J, \ i \in I \quad (83)
\end{align*}
\]

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.

\[
(CRFLP-t) \quad \min \sum_{j} (F_{j} x_{j} + V_{j} c_{j}) + \sum_{s \in S} \sum_{j \in J} \sum_{i \in I} \tau_{s} A_{j,i} D_{i} y_{s,j,i}
\]

s.t. \quad (79) - (83)

\[
y_{s,j,i} - T_{s,j} x_{j} \leq 0 \quad \forall s \in S, \quad j \in J, \quad i \in I
\]  

Data

The data for the instances is taken from Daskin [12]. The original problem considers 49 US cities that simultaneously serve as demand sites and potential distribution centers (DCs). The 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. Originally, demands for a single commodity are assumed to be proportional to the state populations in 1990, and transportation costs are proportional to the great-circle distance between facilities. The original demands and transportation cost are used to generate the first instance; all other instances are generated randomly by sampling demands and transportation costs from uniform distributions bounded between 80 and 120% of the original values.

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 the corresponding probabilities \(<1.2\times10^{-5}\) in all cases). 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 each DC has been left to the value originally used by Snyder and Daskin [40]: \(q = 0.05\).

References

1. Balas, E., Jeroslow, R.G.: Canonical cuts on the unit hypercube. SIAM J. Appl. Math. 23, 61–69 (1972)
2. Barahona, F., Anbil, R.: The volume algorithm: producing primal solutions with a subgradient method. Math. Program. 87, 385–399 (2000)
3. Benders, J.F.: Partitioning procedures for solving mixed-variables programming problems. Numer. Math. 4, 238–252 (1962)
4. Birge, J.R., Louveaux, F.: A multicut algorithm for two-stage stochastic linear programs. Eur. J. Oper. Res. 34, 384–392 (1988)
5. Birge, J.R., Louveaux, F.: Introduction to Stochastic Programming. Springer Series in Operations Research. Springer, New York (2011)
6. Bixby, R., Rothberg, E.: Progress in computational mixed integer programming—a look back from the other side of the tipping point. Ann. Oper. Res. 49, 37–41 (2007)
7. Brooke, A., Kendrick, D., Meeraus, A.: GAMS: A Users Guide, Release 24.2.1. The Scientific Press, South San Francisco (2013)
8. Bussieck, M.R., Ferris, M.C., Meeraus, A.: Grid-enabled optimization with GAMS. INFORMS J. Comput. 21, 349–362 (2009)
9. Caroe, C.C., Schultz, R.: Dual decomposition in stochastic integer programming. Oper. Res. Lett. 24, 37–45 (1999)
10. Cerisola, S., Baillo, A., Fernandez-Lopez, J.M., Ramos, A., Gollmer, R.: Stochastic power generation unit commitment in electricity markets: a novel formulation and a comparison of solution methods. Oper. Res. 57, 32–46 (2009)
11. Cheney, E.W., Goldstein, A.A.: Newton’s method for convex programming and Tchebycheff approximation. Numer. Math. 1, 253–268 (1959)
12. Daskin, M.S.: Network and Discrete Location: Models, Algorithms, and Applications. Wiley, New York (1995). Chap. Appendix H. Longitudes, Latitudes, Demands, and Fixed Cost for SORTCAP.GRT: A 49-Node Problem Defined on the Continental United States
13. Fisher, M.L.: The Lagrangian relaxation method for solving integer programming problems. Manag. Sci. 27, 1–18 (1981)
14. Frangioni, A.: Generalized bundle methods. SIAM J. Optim. 13, 117–156 (2002)
15. Frangioni, A.: About Langrangian methods in integer optimization. Ann. Oper. Res. 139, 163–193 (2005)
16. Garcia-Herreros, P., Wassick, J., Grossmann, I.E.: Design of resilient supply chains with risk of facility disruptions. Ind. Eng. Chem. Res. 53, 17,240–17,251 (2014)
17. Geoffrion, A.M.: Generalized Benders decomposition. J. Optim. Theory Appl. 10, 237–260 (1972)
18. Goffin, J.L., Haurie, A., Vial, J.P.: Decomposition and nondifferentiable optimization with the projective algorithm. Manag. Sci. 38, 284–302 (1992)
19. Guignard, M.: Lagrangean relaxation. Top 11, 151–228 (2003)
20. Guignard, M., Kim, S.: Lagrangean decomposition—a model yielding stronger Lagrangean bounds. Math. Program. 39, 215–228 (1987)
21. Gupta, V.: Modeling and computational strategies for optimal oilfield development planning under fiscal rules and endogenous uncertainties. PhD thesis, Carnegie Mellon University, Pittsburgh (2013)
22. Heitsch, H., Romisch, W.: Scenario reduction algorithms in stochastic programming. Comput. Optim. Appl. 24, 187–206 (2003)
23. Held, M., Karp, R.M.: The traveling-salesman problem and minimum spanning trees: part ii. Math. Program. 1, 6–25 (1971)
24. Held, M., Wolfe, P., Crowder, H.P.: Validation of subgradient optimization. Math. Program. 6, 62–88 (1974)
25. Holmberg, K.: On the convergence of cross decomposition. Math. Program. 47, 269–296 (1990)
26. Holmberg, K.: Linear mean value cross decomposition: a generalization of the Kornai–Liptak method. Eur. J. Oper. Res. 62, 55–73 (1992)
27. Holmberg, K.: A convergence proof for linear mean value cross decomposition. Zeitschrift für Op. Res. 39(2), 157–186 (1994)
28. Holmberg, K.: Mean value cross decomposition applied to integer programming problems. Eur. J. Oper. Res. 97, 124–138 (1997)
29. Karuppiah, R., Grossmann, I.E.: A Lagrangean based branch-and-cut algorithm for global optimization of nonconvex mixed-integer nonlinear programs with decomposable structures. J. Glob. Optim. 41, 163–186 (2008)
30. Kelley, J.E.: The cutting-plane method for solving convex programs. J. Soc. Ind. Appl. Math. 8, 703–712 (1960)
31. Kiwiel, K.C.: Proximity control in bundle methods for convex nondifferentiable minimization. Math. Program. 46, 105–122 (1990)
32. Lemarechal, C.: An algorithm for minimizing convex functions. In: Rosenfeld, J.L. (ed.) Proceedings IFIP74, pp. 552–556, Stockholm (1974)
33. Lima, R., Grossmann, I.E.: Computational advances in solving mixed integer linear programming problems. In: Chemical engineering greetings to Prof. Sauro Pierucci. AIDAC, Milano, Italy, pp. 151–160 (2011)
34. Linderoth, J., Shapiro, A., Wright, S.: The empirical behavior of sampling methods for stochastic programming. Ann. Oper. Res. 142, 215–241 (2006)
35. Magnanti, T.L., Wong, R.T.: Accelerating Benders decomposition: algorithmic enhancement and model selection criteria. Oper. Res. 29, 464–484 (1981)
36. Mouret, S., Grossmann, I.E., Pestiaux, P.: A new Lagrangian decomposition approach applied to the integration of refinery planning and crude-oil scheduling. Comput. Chem. Eng. 35, 2750–2766 (2011)
37. Oliveira, F., Gupta, V., Hamacher, S., Grossmann, I.E.: A Lagrangean decomposition approach for oil supply chain investment planning under uncertainty with risk considerations. Comput. Chem. Eng. 50, 184–195 (2013)
38. Saharidis, G.K.D., Minoux, M., Ierapetritou, M.G.: Accelerating Benders method using covering cut bundle generation. Int. Trans. Oper. Res. 17, 221–237 (2010)
39. Sahinidis, N.V., Grossmann, I.E.: Convergence properties of generalized Benders decomposition. Comput. Chem. Eng. 15, 481–491 (1991)
40. Snyder, L.V., Daskin, M.S.: Reliability models for facility location: the expected failure cost case. Transp. Sci. 39(3), 400–416 (2005)
41. Sohn, H.S., Bricker, D.L., Tseng, T.L.: Mean value cross decomposition for two-stage stochastic linear programming with recourse. Open Oper. Res. J. 5, 30–38 (2011)
42. Tarhan, B.: Stochastic programming approaches for decision-dependent uncertainty and gradual uncertainty resolution. Ph.D thesis, Carnegie Mellon University, Pittsburgh (2009)
43. Van Roy, T.J.: Cross decomposition for mixed integer programming. Math. Program. 25, 46–63 (1983)
44. Van Slyke, R., Wets, R.J.B.: L-shaped linear programs with applications to optimal control and stochastic programming. SIAM J. Appl. Math. 17, 638–663 (1969)
45. You, F., Grossmann, I.E.: Multicut Benders decomposition algorithm for process supply chain planning under uncertainty. Ann. Oper. Res. 210, 191–211 (2011)
46. Zowe, J.: Nondifferentiable optimization. In: Schittkowski, K. (ed.) Computational Mathematical Program, NATO ASI Series F: Computer and Systems Science, vol. 15, pp. 323–356 (1985)