FINITELY CONVERGENT DECOMPOSITION ALGORITHMS FOR TWO-STAGE STOCHASTIC PURE INTEGER PROGRAMS *

MINJIAO ZHANG † AND SİMGE KÜÇÜKYAVUZ ‡

Abstract. We study a class of two-stage stochastic integer programs with general integer variables in both stages and finitely many realizations of the uncertain parameters. Based on Benders' method, we propose a decomposition algorithm that utilizes Gomory cuts in both stages. The Gomory cuts for the second-stage scenario subproblems are parameterized by the first-stage decision variables, i.e., they are valid for any feasible first-stage solutions. In addition, we propose an alternative implementation that incorporates Benders' decomposition into a branch-and-cut process in the first stage. We prove the finite convergence of the proposed algorithms. We also report our preliminary computations with a rudimentary implementation of our algorithms to illustrate their effectiveness.

Key words. Two-stage stochastic pure integer programs, Gomory cuts, Benders' decomposition

1. Introduction.

We investigate a class of two-stage stochastic pure integer programs (SIP) with general integer variables in both stages. We assume that the uncertain data follow a finite discrete distribution, where each realization of the uncertain data is referred to as a scenario. Before the uncertainty is revealed (in the first stage), the decision maker makes strategic decisions. After the uncertain parameters are revealed (in the second stage), the decision maker makes operational decisions in response to the realization of the uncertain parameters to optimize an objective. The typical objective function includes the first-stage cost and the expected second-stage cost.

Let $\tilde{\omega}$ be a random vector defined on a probability space $(\Omega, \mathcal{F}, \mathcal{P})$. Consider the following SIP with the first-stage variables $\bar{x} := (\bar{x}_1, \ldots, \bar{x}_{n_1}) \in \mathbb{Z}_+^{n_1}$ and the second-stage variables $y := (y_0, y_1, \ldots, y_{n_2}) \in \mathbb{Z} \times \mathbb{Z}_+^{n_2}$:

$$\min \quad \bar{c}^\top \bar{x} + \mathbb{E}_{\tilde{\omega}}[f(\bar{x}, \tilde{\omega})] \tag{1.1}$$

$$\text{s.t.} \quad \bar{A}\bar{x} \leq \bar{b}, \tag{1.2}$$

$$\bar{x} \in \mathbb{Z}_+^{n_1}, \tag{1.3}$$

where for a realization $\tilde{\omega} = \omega \in \Omega$, $f(\bar{x}, \tilde{\omega})$ is defined as

$$f(\bar{x}, \omega) = \min \quad y_0 \tag{1.4}$$

$$\text{s.t.} \quad W(\omega)y \leq r(\omega) - \bar{T}(\omega)\bar{x}, \tag{1.5}$$

$$y \in \mathbb{Z} \times \mathbb{Z}_+^{n_2}. \tag{1.6}$$

*This material is based upon work supported by the National Science Foundation under Grant No. 1100383. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

†Department of Information Systems, Statistics, and Management Science, The University of Alabama, 361 Stadium Drive, Tuscaloosa, AL 35487 (mzhang@cba.ua.edu)

‡Corresponding author. Department of Integrated Systems Engineering, The Ohio State University, 1971 Neil Avenue, 210 Baker Systems Building, Columbus, OH 43210 (kucukyavuz.2@osu.edu)

Here, $\bar{c} \in \mathbb{Q}^{n_1}$, $\bar{A} \in \mathbb{Q}^{a \times n_1}$, $\bar{b} \in \mathbb{Q}^a$, the technology matrix $\bar{T}(\omega) \in \mathbb{Q}^{t(\omega) \times n_1}$, the recourse matrix $W(\omega) \in \mathbb{Q}^{t(\omega) \times (n_2+1)}$, and the second-stage right-hand-side vector $r(\omega) \in \mathbb{Q}^{t(\omega)}$ for $\omega \in \Omega$, where a is the number of constraints in the first-stage problem and $t(\omega)$ is the number of constraints in the second-stage problem for $\omega \in \Omega$. Constraints (1.5) include $\sum_{i=1}^{n_2} g_i(\omega) y_i - y_0 \leq 0$, where $g(\omega) \in \mathbb{Z}^{n_2}$ is the vector of cost coefficients of the second-stage decision variables $\{y_i\}_{i=1}^{n_2}$ and y_0 represents the optimal objective function value of the second-stage problem.

Let $\bar{X} = \{\bar{x} : (1.2) - (1.3)\}$ and $Y(\bar{x}, \omega) = \{y : (1.5) - (1.6)\}$. We make the following assumptions:

(A1) $\bar{c}$, $\bar{A}$, $\bar{b}$, $\bar{T}(\omega)$, $W(\omega)$, $r(\omega)$, $g(\omega)$ are integral.
(A2) $\bar{X}$ is nonempty.
(A3) There does not exist an extreme ray r of the polyhedron $\{\bar{x} \in \mathbb{R}_+^{n_1} : \bar{A}\bar{x} \leq \bar{b}\}$ such that $\bar{c}^\top r < 0$. Also, $|f(\bar{x}, \omega)| < +\infty$ for any $(\bar{x}, \omega) \in \bar{X} \times \Omega$.
(A4) $Y(\bar{x}, \omega) \neq \emptyset$ for any $(\bar{x}, \omega) \in \bar{X} \times \Omega$.
(A5) Ω is finite, where $m := |\Omega|$.

Assumption (A1) is made without loss of generality because we can scale these rational parameters by appropriate multipliers to obtain integers. Assumption (A2) makes sure that there exists at least one feasible solution $\bar{x} \in \bar{X}$. Assumption (A3) guarantees that both the first-stage and the second-stage objective functions are bounded. Hence, the objective function of the SIP is bounded. Assumption (A4), known as the relatively complete recourse property, ensures that there exists a feasible solution to the second-stage problems for a feasible first-stage solution.

In addition, from Assumption (A5), we let ω_i denote the i-th realization (scenario) of $\tilde{\omega}$ with $i = 1, \ldots, m$. Let $p_\omega := \mathcal{P}(\tilde{\omega} = \omega) \in [0,1] \cap \mathbb{Q}$ denote the probability of the realization $\tilde{\omega} = \omega \in \Omega$, where $\sum_{\omega \in \Omega} p_\omega = 1$. (Note that there exists a constant $Q \in \mathbb{Z}_+$ such that $Qp_\omega \in \mathbb{Z}_+$ for all $\omega \in \Omega$.) The deterministic equivalent formulation (DEF) for the SIP problem is given by:

$$\text{(1.7)} \qquad \min \quad \bar{c}^\top \bar{x} + \sum_{\omega \in \Omega} p_\omega y_0(\omega)$$

$$\text{(1.8)} \qquad \text{s.t.} \quad \bar{A}\bar{x} \leq \bar{b},$$

$$\text{(1.9)} \qquad \bar{T}(\omega)\bar{x} + W(\omega)y(\omega) \leq r(\omega) \qquad \omega \in \Omega,$$

$$\text{(1.10)} \qquad \bar{x} \in \mathbb{Z}_+^{n_1},$$

$$\text{(1.11)} \qquad y(\omega) \in \mathbb{Z} \times \mathbb{Z}_+^{n_2} \qquad \omega \in \Omega.$$

We introduce two additional variables x_{n_1+1} and x_{n_1+2} to represent the second-stage value function by $x_{n_1+1} - x_{n_1+2}$ (after scaling with Q). Note that if the second-stage value function is known to be nonnegative, then we do not need the variable x_{n_1+2}. Let $x^\top := (\bar{x}^\top, x_{n_1+1}, x_{n_1+2}) \in \mathbb{Z}_+^{n_1+2}$ and $c^\top := (Q\bar{c}^\top, 1, -1)$. Consider the equivalent linear program (LP), after scaling the objective with $Q > 0$, $\min\{c^\top x : (\bar{x}, \{y(\omega)\}_{\omega \in \Omega}) \in \text{conv}((1.8)-(1.11)), x_{n_1+1} - x_{n_1+2} = \sum_{\omega \in \Omega} Qp_\omega y_0(\omega)\}$. Because the feasible region of this LP is non-empty and the objective function (1.7) is bounded from below (Assumption A3), according to Theorem 2.2 in [16], there exists a set of constraints $(x, \{y(\omega)\}_{\omega \in \Omega}) \leq M$ that when added to the formulation does not cut off the optimal solution, where M is an $(n_1+2+m(n_2+1))$-dimensional vector of finite constants. This result allows us to assume that the upper bound constraints on $\bar{x}$ and $y(\omega), \omega \in \Omega$ are present in the constraints (1.2) and (1.5), respectively. Hence, we make the assumption, without loss of generality,

(A6) The sets $\bar{X}$ and $Y(\bar{x},\omega)$ for any $(\bar{x},\omega) \in \bar{X} \times \Omega$, and the variables x_{n_1+1}, x_{n_1+2} are bounded.

In this paper, we propose a decomposition algorithm based on Benders' [3] and L-shaped methods [23] to solve this very large-scale pure integer program. Our algorithm solves the first-stage problem and the multiple second-stage problems for each scenario as linear programs, and it utilizes Gomory cuts [10] to convexify the first- and second-stage problems. The proposed algorithm has many attractive features, including its applicability to two-stage integer programs with random recourse and technology matrices, and cost and right-hand-side vectors, as well as its use of optimality cuts that are affine in the first-stage general integer variables. In addition, the decomposition algorithm naturally lends itself to parallelization. We also give an alternative implementation, which solves the first-stage problem using a branch-and-cut algorithm. Our preliminary computational experience is encouraging.

1.1. Literature Review. In this section, we give a brief overview of related research in two-stage stochastic mixed-integer programming. For a more detailed survey on various algorithms for stochastic mixed-integer programming, we refer the reader to Sen [18].

Laporte and Louveaux [14] propose the L-shaped decomposition algorithm for two-stage stochastic programs with binary variables in the first stage and mixed-integer variables in the second stage. This algorithm requires the solution of the second-stage mixed-integer programs to optimality in each iteration. For problems with mixed 0-1 (binary and continuous) variables in both stages, Carøe and Tind [6] propose a method to update the lift-and-project cuts [2] generated from one scenario to be valid for all other scenarios. Sen and Higle [19] develop a decomposition algorithm for the stochastic integer programs with binary variables in the first stage and mixed 0-1 variables in the second-stage. This method involves generating disjunctive cuts to convexify the master problem and scenario subproblems. Sen and Sherali [20] develop an extension of this algorithm that involves branch-and-cut algorithm to solve the second-stage mixed-integer programs. Both of these algorithms utilize the assumption that the recourse matrix is fixed. Sherali and Zhu [21] develop a decomposition-based branch-and-bound algorithm based on a hyperrectangular partitioning process, which relies on the restriction that the second-stage variables are binary or the first-stage variables are extreme points of the hyperrectangular space.

For two-stage stochastic programs with mixed-integer variables in the first stage and general integer variables in the second stage, Carøe and Tind [7] propose a conceptual method that solves the second-stage program for a given first-stage solution to integer optimality by iteratively adding Gomory cuts, and constructs the optimality cuts that are in terms of a series Chvátal functions, which are non-convex [4]. Ahmed et al. [1] develop a finite terminating branch-and-bound method by reformulating this class of problems with the assumption that the technology matrix is fixed. If the technology matrix is dependent on the scenario, then this method needs to branch on the number of scenarios times more variables, which results in exponentially more iterations. In another line of work, Schultz et al. [17] consider the case with continuous variables in the first stage, integer variables in the second stage, and fixed technology and recourse matrices. They propose a method that enumerates all possible optimal solutions, which are contained in a countable set.

With the assumption that the decision variables are pure integers in both stages, Kong et al. [12] propose an equivalent superadditive dual formulation and use a branch-and-bound or level-set approach to find the optimal solution. Also, Trapp et al. [22] develop an algorithmic framework based on the characterization of the value function by level-sets. However, both Kong et al. [12] and Trapp et al. [22] assume that either the second-stage cost function or the technology and recourse matrices are fixed, i.e., they are not affected by the random parameters. In contrast, in this paper, we allow all these data to be random.

One of the most relevant work to this paper is Gade et al. [9], who develop a decomposition algorithm for two-stage stochastic programs with *binary* first-stage decisions and integer second-stage decisions. They allow the second-stage cost function, technology and recourse matrices to be random. However, because this decomposition method exploits the property that the first-stage variables are binary to derive valid cuts for the second stage, it is not directly extendable to stochastic integer programs with general integer variables in the first stage.

In addition to the primal decomposition methods referenced, there is also a class of algorithms based on dual decomposition. Carøe and Schultz [5] propose a branch-and-bound scheme for the two-stage stochastic programs with mixed-integer variables in both stages, in which the Lagrangian dual obtained by dualizing the non-anticipativity constraints provides a lower bound on the optimal objective value. Based on Carøe and Schultz [5], Lubin et al. [15] develop a formulation that allows a parallelized solution of the master problem.

2. A Decomposition Algorithm with Parametric Gomory Cuts. In this section, we develop a decomposition algorithm with Gomory cuts to solve the two-stage SIP. Our overall approach is to utilize Benders' decomposition algorithm to iteratively add optimality cuts to the first-stage problem to approximate the second-stage value function. In each iteration, Gomory cuts are generated after solving the linear relaxations of the first-stage and second-stage subproblems for each scenario separately.

2.1. Parametric Gomory Cuts. Gomory [10] proposes a class of inequalities and a pure cutting plane algorithm for deterministic pure integer programs. Suppose that the decision variables $z \in \mathbb{Z}_+^n$ satisfy a constraint $\sum_{i=1}^n \delta_i z_i = b_0$, where $\delta \in \mathbb{R}^n$ and $b_0 \in \mathbb{R}$. Then the inequality $\sum_{i=1}^n \lfloor \delta_i \rfloor z_i \leq \sum_{i=1}^n \delta_i z_i = b_0$ is valid, and because

$\{\lfloor\delta_i\rfloor\}_{i=1}^n \in \mathbb{Z}^n$ and $z \in \mathbb{Z}_+^n$, then $\sum_{i=1}^n \lfloor\delta_i\rfloor z_i \in \mathbb{Z}$. The resulting Gomory cut is

$$\sum_{i=1}^{n} \lfloor\delta_i\rfloor z_i \leq \lfloor b_0\rfloor. \tag{2.1}$$

In solving a pure integer program with Gomory's cutting plane method, we solve its linear relaxation and generate a Gomory cut in each iteration to cut off a fractional solution. Gomory [10] shows that the optimal integer solution can be found using this pure cutting plane method in finitely many iterations, when the linear programs are solved using the lexicographic dual simplex method and the Gomory cut is generated from the fractional variable with the smallest index. (See also [8].)

Given a particular first-stage solution, $\bar{x}$, the Gomory cut obtained as we solve the linear relaxation of the second-stage subproblem for $\omega \in \Omega$ is not necessarily valid for all other feasible first-stage solutions. Our first goal is to develop a Gomory cut $\pi(\omega)^\top y(\omega) \leq \pi_0(\bar{x}, \omega)$ that is valid for the linear relaxation of the DEF, where $\pi(\omega) \in \mathbb{Z}^{n_2+1}$ and $\pi_0(\bar{x}, \omega)$ is an affine function of the first-stage decisions $\bar{x}$.

For purposes of utilizing the simplex method and deriving Gomory cuts, we redefine matrices $\bar{A}$, $\bar{T}(\omega)$, and $W(\omega)$ to include the slack variables in both stages. For a given optimal solution to the linear relaxation of the first-stage problem, let $\bar{A}_{B_1} = [\bar{A}_{B_1(1)}, \ldots, \bar{A}_{B_1(a)}]$ denote the corresponding basis matrix for the first-stage problem, in which $B_1(1), \ldots, B_1(a)$ are the indices of the columns in the basis matrix and B_1 stands for basis for the first-stage problem. Denote $\bar{x}_{B_1} = (\bar{x}_{B_1(1)}, \ldots, \bar{x}_{B_1(a)})$ as the first-stage basic variables. Note that $\bar{x}_{B_1} = \bar{A}_{B_1}^{-1}\bar{b}$. In addition, let $\bar{T}_{B_1}(\omega) = [\bar{T}_{B_1(1)}(\omega), \ldots, \bar{T}_{B_1(a)}(\omega)]$, $\omega \in \Omega$ be defined similarly. Then we solve the linear relaxation of the second-stage problem for the given $\bar{x}$, and let $W_{B_2}(\omega)$ denote the corresponding basis matrix to the second-stage problem for $\omega \in \Omega$, where B_2 stands for basis for the second-stage problem. Note that B_2 is dependent on ω, but we drop this dependence for notational convenience. Then the second-stage basic variables $y_{B_2}(\omega_i) = W_{B_2}(\omega_i)^{-1}(r(\omega_i) - \bar{T}(\omega_i)\bar{x})$ for $\omega_i \in \Omega$.

LEMMA 2.1. $\begin{bmatrix} \bar{A}_{B_1} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \bar{T}_{B_1}(\omega_1) & W_{B_2}(\omega_1) & \mathbf{0} & \cdots & \mathbf{0} \\ \bar{T}_{B_1}(\omega_2) & \mathbf{0} & W_{B_2}(\omega_2) & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \bar{T}_{B_1}(\omega_m) & \mathbf{0} & \mathbf{0} & \cdots & W_{B_2}(\omega_m) \end{bmatrix}$ *is a feasible basis matrix for DEF.*

Proof. First, because the columns of the matrices $\bar{A}_{B_1}$, $W_{B_2}(\omega_1)$, ..., $W_{B_2}(\omega_m)$ are linearly independent, clearly the matrix $\begin{bmatrix} \bar{A}_{B_1} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \bar{T}_{B_1}(\omega_1) & W_{B_2}(\omega_1) & \mathbf{0} & \cdots & \mathbf{0} \\ \bar{T}_{B_1}(\omega_2) & \mathbf{0} & W_{B_2}(\omega_2) & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \bar{T}_{B_1}(\omega_m) & \mathbf{0} & \mathbf{0} & \cdots & W_{B_2}(\omega_m) \end{bmatrix}$ is full rank.

To show the feasibility of the basis matrix, we also need to prove that

$$\begin{bmatrix} \bar{A}_{B_1} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \bar{T}_{B_1}(\omega_1) & W_{B_2}(\omega_1) & \mathbf{0} & \cdots & \mathbf{0} \\ \bar{T}_{B_1}(\omega_2) & \mathbf{0} & W_{B_2}(\omega_2) & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \bar{T}_{B_1}(\omega_m) & \mathbf{0} & \mathbf{0} & \cdots & W_{B_2}(\omega_m) \end{bmatrix}^{-1} \begin{bmatrix} \bar{b} \\ r(\omega_1) \\ r(\omega_2) \\ \vdots \\ r(\omega_m) \end{bmatrix} \geq \mathbf{0}.$$

Because $\bar{x}_{B_1} = \bar{A}_{B_1}^{-1}\bar{b} \geq 0$ and $y_{B_2}(\omega_i) = W_{B_2}(\omega_i)^{-1}(r(\omega_i) - \bar{T}_{B_1}(\omega_i)\bar{x}_{B_1}) \geq 0$ for $i = 1, \ldots, m$, we have

$$\begin{aligned} &\begin{bmatrix} \bar{A}_{B_1} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \bar{T}_{B_1}(\omega_1) & W_{B_2}(\omega_1) & \mathbf{0} & \cdots & \mathbf{0} \\ \bar{T}_{B_1}(\omega_2) & \mathbf{0} & W_{B_2}(\omega_2) & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \bar{T}_{B_1}(\omega_m) & \mathbf{0} & \mathbf{0} & \cdots & W_{B_2}(\omega_m) \end{bmatrix}^{-1} \begin{bmatrix} \bar{b} \\ r(\omega_1) \\ r(\omega_2) \\ \vdots \\ r(\omega_m) \end{bmatrix} \\ &= \begin{bmatrix} \bar{A}_{B_1}^{-1} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ -W_{B_2}(\omega_1)^{-1}\bar{T}_{B_1}(\omega_1)\bar{A}_{B_1}^{-1} & W_{B_2}(\omega_1)^{-1} & \mathbf{0} & \cdots & \mathbf{0} \\ -W_{B_2}(\omega_2)^{-1}\bar{T}_{B_1}(\omega_2)\bar{A}_{B_1}^{-1} & \mathbf{0} & W_{B_2}(\omega_2)^{-1} & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -W_{B_2}(\omega_m)^{-1}\bar{T}_{B_1}(\omega_m)\bar{A}_{B_1}^{-1} & \mathbf{0} & \mathbf{0} & \cdots & W_{B_2}(\omega_m)^{-1} \end{bmatrix} \begin{bmatrix} \bar{b} \\ r(\omega_1) \\ r(\omega_2) \\ \vdots \\ r(\omega_m) \end{bmatrix} \\ &= \begin{bmatrix} \bar{A}_{B_1}^{-1}\bar{b} \\ -W_{B_2}(\omega_1)^{-1}\bar{T}_{B_1}(\omega_1)\bar{A}_{B_1}^{-1}\bar{b} + W_{B_2}(\omega_1)^{-1}r(\omega_1) \\ -W_{B_2}(\omega_2)^{-1}\bar{T}_{B_1}(\omega_2)\bar{A}_{B_1}^{-1}\bar{b} + W_{B_2}(\omega_2)^{-1}r(\omega_2) \\ \vdots \\ -W_{B_2}(\omega_m)^{-1}\bar{T}_{B_1}(\omega_m)\bar{A}_{B_1}^{-1}\bar{b} + W_{B_2}(\omega_m)^{-1}r(\omega_m) \end{bmatrix} = \begin{bmatrix} \bar{x}_{B_1} \\ y_{B_2}(\omega_1) \\ y_{B_2}(\omega_2) \\ \vdots \\ y_{B_2}(\omega_n) \end{bmatrix} \geq \mathbf{0}. \end{aligned}$$

□

For given first-stage basis matrix $\bar{A}_{B_1}$, second-stage basis matrices $W_{B_2}(\omega)$, and the submatrices $\bar{T}_{B_1}(\omega)$ for $\omega \in \Omega$, let

$$G := \begin{bmatrix} \bar{A}_{B_1}^{-1} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ -W_{B_2}(\omega_1)^{-1}\bar{T}_{B_1}(\omega_1)\bar{A}_{B_1}^{-1} & W_{B_2}(\omega_1)^{-1} & \mathbf{0} & \cdots & \mathbf{0} \\ -W_{B_2}(\omega_2)^{-1}\bar{T}_{B_1}(\omega_2)\bar{A}_{B_1}^{-1} & \mathbf{0} & W_{B_2}(\omega_2)^{-1} & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -W_{B_2}(\omega_m)^{-1}\bar{T}_{B_1}(\omega_m)\bar{A}_{B_1}^{-1} & \mathbf{0} & \mathbf{0} & \cdots & W_{B_2}(\omega_m)^{-1} \end{bmatrix}.$$

Then the Gomory cuts generated from any row of

$$G\begin{bmatrix}\bar{A}\\ \bar{T}(\omega_1)\\ \bar{T}(\omega_2)\\ \vdots\\ \bar{T}(\omega_m)\end{bmatrix}\bar{x}+G\begin{bmatrix}\mathbf{0} & \mathbf{0} & \cdots & \mathbf{0}\\ W(\omega_1) & \mathbf{0} & \cdots & \mathbf{0}\\ \mathbf{0} & W(\omega_2) & \cdots & \mathbf{0}\\ \vdots & \vdots & \ddots & \vdots\\ \mathbf{0} & \mathbf{0} & \cdots & W(\omega_m)\end{bmatrix}y=G\begin{bmatrix}\bar{b}\\ r(\omega_1)\\ r(\omega_2)\\ \vdots\\ r(\omega_m)\end{bmatrix} \tag{2.2}$$

are valid for DEF. Such cuts are referred to as *parametric Gomory cuts* in the rest of this paper, because they are parameterized with respect to the first-stage decision variables $\bar{x}$. We demonstrate these cuts on an instance that is adapted from the test set 1 in Ahmed et al. [1]. Throughout the paper, we let $[i,j] := \{i, i+1, \dots, j\}$ for $i, j \in \mathbb{Z}$.

Example 1. *Consider a two-stage SIP with $a = 2$, $t(\omega) = 2$, $p_\omega = \frac{1}{m}, \omega \in \Omega$ and $m = Q = 3$, whose DEF is given by*

$$\begin{aligned}
\min \quad & -6m\bar{x}_1 - 16m\bar{x}_2 - \sum_{i=1}^{m} y_0(\omega_i) \\
\text{s.t.} \quad & \bar{x}_1 \le 5, && (2.3)\\
& \bar{x}_2 \le 5, && (2.4)\\
& y_0(\omega_1) - 17y_1(\omega_1) - 20y_2(\omega_1) - 24y_3(\omega_1) - 28y_4(\omega_1) \le 0, && (2.5)\\
& 3y_1(\omega_1) + 4y_2(\omega_1) + 5y_3(\omega_1) + 5y_4(\omega_1) \le 24 - 2\bar{x}_1, && (2.6)\\
& 7y_1(\omega_1) + y_2(\omega_1) + 4y_3(\omega_1) + 3y_4(\omega_1) \le 23 - 3\bar{x}_2, && (2.7)\\
& y_0(\omega_2) - 17y_1(\omega_2) - 19y_2(\omega_2) - 24y_3(\omega_2) - 28y_4(\omega_2) \le 0, && (2.8)\\
& 3y_1(\omega_2) + 3y_2(\omega_2) + 4y_3(\omega_2) + 6y_4(\omega_2) \le 27 - 3\bar{x}_1, && (2.9)\\
& 6y_1(\omega_2) + y_2(\omega_2) + 4y_3(\omega_2) + 3y_4(\omega_2) \le 22 - \bar{x}_2, && (2.10)\\
& y_0(\omega_3) - 16y_1(\omega_3) - 19y_2(\omega_3) - 24y_3(\omega_3) - 29y_4(\omega_3) \le 0, && (2.11)\\
& 2y_1(\omega_3) + 3y_2(\omega_3) + 4y_3(\omega_3) + 6y_4(\omega_3) \le 29 - 4\bar{x}_1, && (2.12)\\
& 6y_1(\omega_3) + 2y_2(\omega_3) + 4y_3(\omega_3) + 3y_4(\omega_3) \le 23 - 4\bar{x}_2, && (2.13)\\
& \bar{x} \in \mathbb{Z}^2_+, \\
& y(\omega_i) \in \mathbb{Z} \times \mathbb{Z}^4_+, \quad i \in [1,3].
\end{aligned}$$

First, we introduce the slack variables $\bar{x}_3$, $\bar{x}_4$, $\{y_5(\omega_i)\}_{i\in[1,3]}$, $\{y_6(\omega_i)\}_{i\in[1,3]}$, and $\{y_7(\omega_i)\}_{i\in[1,3]}$ to put the problem in standard form. Then, we solve the linear relaxation of the first-stage problem

$$\begin{aligned}
\min \quad & -18\bar{x}_1 - 48\bar{x}_2 \\
\text{s.t} \quad & \bar{x}_1 + \bar{x}_3 = 5, \\
& \bar{x}_2 + \bar{x}_4 = 5, \\
& \bar{x} \in \mathbb{R}^4_+,
\end{aligned}$$

by lexicographic simplex method. The optimal tableau is

	$\bar{x}_1$	$\bar{x}_2$	$\bar{x}_3$	$\bar{x}_4$
330	0	0	18	48
5	1	0	1	0
5	0	1	0	1

Thus $\bar{A}_{B_1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $\bar{A}_{B_1}^{-1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, *and the optimal solution is* $\bar{x} = (5, 5, 0, 0)$. *Then we solve the linear relaxation of the second-stage subproblem for given* $\bar{x}$ *for the first scenario by lexicographic simplex method. The optimal tableau is*

	$y_0(\omega_1)$	$y_1(\omega_1)$	$y_2(\omega_1)$	$y_3(\omega_1)$	$y_4(\omega_1)$	$y_5(\omega_1)$	$y_6(\omega_1)$	$y_7(\omega_1)$
77.71	0	8.71	0	5.71	0	1	4.57	1.71
77.71	1	8.71	0	5.71	0	1	4.57	1.71
0.29	0	-3.71	1	-0.71	0	0	0.43	-0.71
2.57	0	3.57	0	1.57	1	0	-0.14	0.57

Here, $W_{B_2}(\omega_1) = \begin{bmatrix} 1 & -20 & -28 \\ 0 & 4 & 5 \\ 0 & 1 & 3 \end{bmatrix}$, *and* $\bar{T}_{B_1}(\omega_1) = \begin{bmatrix} 0 & 0 \\ 2 & 0 \\ 0 & 3 \end{bmatrix}$.

Therefore, $\begin{bmatrix} \bar{A}_{B_1}^{-1} & \mathbf{0} \\ -W_{B_2}(\omega_1)^{-1}\bar{T}_{B_1}(\omega_1)\bar{A}_{B_1}^{-1} & W_{B_2}(\omega_1)^{-1} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ -9.14 & -5.14 & 1 & 4.57 & 1.71 \\ -0.86 & 2.14 & 0 & 0.43 & -0.71 \\ 0.29 & -1.71 & 0 & -0.14 & 0.57 \end{bmatrix}$.

Consider the source row corresponding to $y_0(\omega_1)$*:*

$$\begin{bmatrix} -9.14 \\ -5.14 \\ 1 \\ 4.57 \\ 1.71 \end{bmatrix}^{\top} \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -17 & -20 & -24 & -28 & 1 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 3 & 4 & 5 & 5 & 0 & 1 & 0 \\ 0 & 3 & 0 & 0 & 0 & 7 & 1 & 4 & 3 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \bar{x} \\ y(\omega_1) \end{bmatrix} = \begin{bmatrix} -9.14 \\ -5.14 \\ 1 \\ 4.57 \\ 1.71 \end{bmatrix}^{\top} \begin{bmatrix} 5 \\ 5 \\ 0 \\ 24 \\ 23 \end{bmatrix}.$$

The Gomory cut obtained from this row is $6y_1(\omega_1) + 3y_2(\omega_1) + 5y_3(\omega_1) + 5y_4(\omega_1) \leq 38 - 2\bar{x}_1 - 3\bar{x}_2$ *after substituting out the slack variables. This cut is valid for the second-stage problem for* ω_1 *for any given* $\bar{x}$.

If we generate the Gomory cut directly from the source row $y_0(\omega_1) + 8.71y_1(\omega_1) + 5.71y_3(\omega_1) + y_5(\omega_1) + 4.57y_6(\omega_1) + 1.71y_7(\omega_1) = 77.71$ *in the second-stage optimal tableau, then the Gomory cut obtained is*

$$6y_1(\omega_1) + 3y_2(\omega_1) + 5y_3(\omega_1) + 5y_4(\omega_1) \leq 13, \tag{2.14}$$

after substituting out $y_5(\omega_1)$, $y_6(\omega_1)$, *and* $y_7(\omega_1)$. *However, this inequality is not necessarily valid for other* $\bar{x} \in \bar{X}$. *For example, for* $\bar{x} = (0, 0, 5, 5) \in \bar{X}$, *the constraints* (2.6)-(2.7) *are reduced to* $3y_1(\omega_1) + 4y_2(\omega_1) + 5y_3(\omega_1) + 5y_4(\omega_1) + y_6(\omega_1) = 24$ *and* $7y_1(\omega_1) + y_2(\omega_1) + 4y_3(\omega_1) + 3y_4(\omega_1) + y_7(\omega_1) = 23$. *The second stage solution* $y(\omega_1) = (82, 2, 0, 2, 0) \in \{y(\omega_1) \in \mathbb{Z}^5 : (2.5) - (2.7), \bar{x} = (0, 0, 5, 5)\}$ *violates inequality*

(2.14), *hence inequality* (2.14) *is not valid for* $\bar{x} = (0, 0, 5, 5)$.

Next, we develop a decomposition algorithm using parametric Gomory cuts to solve two-stage pure SIPs.

2.2. A Cutting Plane Based Decomposition Algorithm. Let θ be a known lower bound of the second-stage value function. Let $T(\omega) := \begin{bmatrix}\bar{T}(\omega) \ \mathbf{0}_{t(\omega)\times 1} \ \mathbf{0}_{t(\omega)\times 1}\end{bmatrix}$ for each $\omega \in \Omega$, and $A := \begin{bmatrix} \bar{A} & \mathbf{0}_{a\times 1} & \mathbf{0}_{a\times 1} \\ \mathbf{0}_{1\times n_1} & -1 & 1 \end{bmatrix}$ and $b := \begin{bmatrix} \bar{b} \\ -\theta \end{bmatrix}$.

First, we define the master problem MP^k for $k \geq 0$ as

$$\begin{aligned} \text{MP}^k: \quad \min \quad & c^\top x \\ \text{s.t.} \quad & A^k x \leq b^k, \\ & x \in \mathbb{R}_+^{n_1+2}, \end{aligned} \tag{2.15}$$

where $A^k x \leq b^k$ includes the original constraints $Ax \leq b$, (i.e., $\bar{A}x \leq \bar{b}$, and the constraint on the second-stage value function $-x_{n_1+1} + x_{n_1+2} \leq -\theta$) and for $k \geq 1$ the Gomory cuts generated for the first-stage problem in iterations $j = 1, \ldots, k$, and the optimality cuts

$$\sum_{\omega\in\Omega} Qp_\omega(\beta^j(\omega))^\top (r^j(\omega) - T^j(\omega)x) \leq x_{n_1+1} - x_{n_1+2} \tag{2.16}$$

generated in iterations $j = 1, \ldots, k$, where $\beta^j(\omega)$ is the optimal dual vector of the subproblem $\text{SP}^j(x, \omega)$. The subproblem $\text{SP}^k(x, \omega)$ for $k \geq 0$ is given by

$$\begin{aligned} \text{SP}^k(x,\omega): \quad f^k(x,\omega) := \min \quad & y_0(\omega) \\ \text{s.t.} \quad & W^k(\omega)y(\omega) \leq r^k(\omega) - T^k(\omega)x, \\ & y(\omega) \in \mathbb{R} \times \mathbb{R}_+^{n_2}, \end{aligned} \tag{2.17}$$

where $W^k(\omega)y(\omega) \leq r^k(\omega) - T^k(\omega)x$ includes the original constraints $W(\omega)y(\omega) \leq r(\omega) - T(\omega)x$, and for $k \geq 1$ the parametric Gomory cuts generated for the second-stage problem for realization ω in iterations 1 to k. We scale inequalities (2.16) and (2.17) so that all coefficients are integral.

Let l_k be the number of rows in matrix A^k ($l_0 := a + 1$), LB and UB be the lower and upper bounds of the optimal objective function value of the DEF. Let $q \in \mathbb{Z}_+$ be the frequency of implementing full Gomory cutting plane method to the master problem. In other words, every q iterations, we implement a pure cutting plane algorithm to solve the master problem to integer optimality. In all other iterations, we solve the linear relaxation of the master problem. (Note that, in these iterations, we could also add one or more violated Gomory cuts.) We denote the optimal solution to the master problem MP^k as x^k and the optimal solution to $\text{SP}^k(x, \omega)$ as $y^k(\omega)$.

Initially, we have $k = 0$, $l_0 = a + 1$, $LB = -\infty$, $UB = +\infty$. If $UB - LB \leq \epsilon$, where ϵ is a very small nonnegative constant, then we have found an optimal integer solution so we stop. Otherwise, $k \leftarrow k + 1$ and we repeat the following process until

$UB - LB \leq \epsilon$.

In iteration k, we first solve the master problem MP^{k-1} by lexicographic dual simplex method (if $k = 1$, use lexicographic simplex method) to obtain the optimal solution x^{k-1}. Then we generate master problem MP^k to be the same as problem MP^{k-1}. Let $x^k = x^{k-1}$. If $x^k \notin \mathbb{Z}_+^{n_1+2}$ and $k \equiv 0 \bmod q$, then we construct a Gomory cut corresponding to the fractional component in x^k with the smallest index, and we update master problem MP^k by updating matrices A^k and b^k with this Gomory cut. Then we re-solve the updated problem MP^k with lexicographic dual simplex method to obtain a new x^k. If $x^k \notin \mathbb{Z}_+^{n_1+2}$ again, then we continue to add Gomory cuts to the master problem and re-solve it with lexicographic dual simplex method until $x^k \in \mathbb{Z}_+^{n_1+2}$. The lower bound LB is updated by the optimal objective function value of MP^k.

Next, we solve the subproblems $\mathrm{SP}^{k-1}(x^k, \omega)$ to obtain $y^{k-1}(\omega)$ for $\omega \in \Omega$. Note that x^k could be fractional when solving the subproblems in iteration $k \not\equiv 0 \bmod q$. We generate subproblem $\mathrm{SP}^k(x, \omega)$ to be the same as subproblem $\mathrm{SP}^{k-1}(x, \omega)$ for each $\omega \in \Omega$. Let $y^k(\omega) = y^{k-1}(\omega)$ for $\omega \in \Omega$. If $y^k(\omega) \notin \mathbb{Z} \times \mathbb{Z}_+^{n_2}$, then we develop a parametric Gomory cut from the fractional component in $y^k(\omega)$ with the smallest index, and update subproblem $\mathrm{SP}^k(x, \omega)$ by updating matrices $T^k(\omega)$, $W^k(\omega)$, and $r^k(\omega)$ with this parametric Gomory cut. Then we solve the updated subproblems $\mathrm{SP}^k(x^k, \omega)$ by lexicographic dual simplex method to obtain new $y^k(\omega)$. If $x^k \in \mathbb{Z}_+^{n_1+2}$ and $y^k(\omega) \in \mathbb{Z} \times \mathbb{Z}_+^{n_2}$ for every $\omega \in \Omega$, then we update the upper bound UB as $\min\{UB, Q(\bar{c}^\top \bar{x}^k + \sum_{\omega \in \Omega} p_\omega f^k(\bar{x}^k, \omega))\}$. In addition, we update the master problem MP^k with the optimality cut (2.16) obtained from the subproblems.

For a given q, the algorithm for solving a SIP problem is given in Algorithm 1, in which Algorithm 2 is for solving the subproblems. Note that if $q = 1$, then Algorithm 1 becomes a pure Gomory cutting plane algorithm for the master problem, and it stops with an integer first-stage solution. In contrast, we solve at most two linear programs and add at most one parametric Gomory cut for each second stage subproblem at every iteration. Although Gomory cutting plane algorithm is finitely convergent, its convergence may be slow for practical purposes. However, solving the master problem to integer optimality with this algorithm at every iteration is not necessary. A potentially more practical implementation is to solve the master problem to integer optimality at every $q > 1$ iterations.

Next we illustrate Algorithm 1 on Example 1.

Example 1. (Continued). Let $q = 1$ and $\epsilon = 0$. In the rest of this example, we let $x = (x_1, x_2, x_3, x_4) := (\bar{x}_1, \bar{x}_2, x_3, x_4)$, where $x_3 - x_4$ represents $-\sum_{i=1}^3 y_0(\omega_i)$. The lower bounding constraint that we use for $x_3 - x_4$ is $x_3 - x_4 \geq -1000$.

Initialization. Let $LB = -\infty$, $UB = +\infty$, $A^0 = A$, $b^0 = b$, $l_0 = 3$, $W^0(\omega_i) = W(\omega_i)$, $T^0(\omega_i) = T(\omega_i)$, $r^0(\omega_i) = r(\omega_i)$, $i \in [1, 3]$.

Iteration 1. $k = 1$. Because $UB - LB > 0$, we solve MP^0 with lexicographic simplex method and obtain $x^0 = (5, 5, 0, 1000)$. Then we generate master problem MP^1, in which $A^1 = A^0$ and $b^1 = b^0$, and let $x^1 = x^0$. We update the lower bound as $LB = -1330$.

Algorithm 1: Algorithm for solving a SIP problem

Initialization: $k \leftarrow 0$, $LB \leftarrow -\infty$, $UB \leftarrow +\infty$;
$A^0 \leftarrow A$, $b^0 \leftarrow b$, $W^0(\omega) \leftarrow W(\omega)$, $T^0(\omega) \leftarrow T(\omega)$, $r^0(\omega) \leftarrow r(\omega)$, $l_0 \leftarrow a+1$;
while $UB - LB > \epsilon$ **do**
 $k \leftarrow k+1$, $l_k \leftarrow l_{k-1}$;
 Solve MP^{k-1} with lexicographic dual simplex to obtain x^{k-1};
 Generate MP^k with $A^k \leftarrow A^{k-1}$ and $b^k \leftarrow b^{k-1}$;
 Let $x^k \leftarrow x^{k-1}$;
 while $x^k \notin \mathbb{Z}_+^{n_1+2}$ *and* $k \equiv 0$ *mod* q **do**
  Generate a Gomory cut from the source row corresponding to $\min\{j : j \in \{1, \ldots, n_1+2\}, x_j^k \notin \mathbb{Z}_+\}$;
  Add the Gomory cut to MP^k, and update A^k and b^k;
  $l_k \leftarrow l_k + 1$;
  Solve MP^k using the lexicographic dual simplex method to obtain x^k;
 end
 $LB \leftarrow c^\top x^k$;
 Let the basis matrix for x^k be $A_{B_1}^k := [A_{B_1(1)}^k, \ldots, A_{B_1(l_k)}^k]$ and $T_{B_1}^k(\omega) := [T_{B_1(1)}^k(\omega), \ldots, T_{B_1(l_k)}^k(\omega)]$ for each $\omega \in \Omega$;
 Input k, UB, c, b^k, x^k, A^k, $A_{B_1}^k$, $W^{k-1}(\omega)$, $T^{k-1}(\omega)$, $T_{B_1}^{k-1}(\omega)$, $r^{k-1}(\omega)$ for all $\omega \in \Omega$ to Algorithm 2;
 Update MP^k with the optimality cut returned from Algorithm 2;
 $l_k \leftarrow l_k + 1$;
end
Return $(\bar{x}^k, \{y^k(\omega)\}_{\omega \in \Omega}), UB, LB$.

We solve subproblems $SP^0(x^1, \omega_i)$, $i \in [1,3]$, with lexicographic simplex method to obtain $y^0(\omega_i)$. Then we generate subproblem $SP^1(x, \omega_i)$ with $i \in [1,3]$, in which $W^1(\omega_i) = W^0(\omega_i)$, $T^1(\omega_i) = T^0(\omega_i)$, and $r^1(\omega_i) = r^0(\omega_i)$. Let $y^1(\omega_i) = y^0(\omega_i), i \in [1,3]$. We have $y^1(\omega_1) = (77.71, 0, 0.29, 0, 2.57) \notin \mathbb{Z}^5$. As demonstrated earlier, we add a parametric Gomory cut

$$6y_1(\omega_1) + 3y_2(\omega_1) + 5y_3(\omega_1) + 5y_4(\omega_1) \leq 38 - 2x_1 - 3x_2 \tag{2.18}$$

to $SP^1(x, \omega_1)$, re-solve $SP^1(x^1, \omega_1)$ and obtain $y^1(\omega_1) = (76, 0, 1, 0, 2) \in \mathbb{Z}^5$. In addition $y^1(\omega_2) = (76, 0, 4, 0, 0)$, which is integral, and $y^1(\omega_3) = (28.5, 0, 1.5, 0, 0) \notin \mathbb{Z}^5$, so we add a parametric Gomory cut

$$3y_1(\omega_3) + y_2(\omega_3) + 2y_3(\omega_3) + y_4(\omega_3) \leq 11 - 2x_2 \tag{2.19}$$

to $SP^1(x, \omega_3)$. Re-solving $SP^1(x^1, \omega_3)$, the new optimal solution we get is $y^1(\omega_3) = (28, 0, 0, 0, 1) \in \mathbb{Z}^5$. Because $x^1 \in \mathbb{Z}^4$ and $y^1(\omega_i) \in \mathbb{Z}^5$, $i \in [1,3]$, we update the upper bound as $UB = -510$. We update the master problem MP^1 with the optimality cut

$$453x_1 + 678x_2 - 15x_3 + 15x_4 \leq 8355. \tag{2.20}$$

Let $l_1 = l_0 + 1 = 4$.

Algorithm 2: Algorithm for solving the subproblems

Given k, UB, c, b^k, x^k, A^k, $A^k_{B_1}$, $W^{k-1}(\omega)$, $T^{k-1}(\omega)$, $T^{k-1}_{B_1}(\omega)$, $r^{k-1}(\omega)$ for all $\omega \in \Omega$;

for $\omega \in \Omega$ **do**

  Solve subproblems $\mathrm{SP}^{k-1}(x^k, \omega)$ using the lexicographic dual simplex method to obtain $y^{k-1}(\omega)$;

  Generate $\mathrm{SP}^k(x, \omega)$ with $W^k(\omega) \leftarrow W^{k-1}(\omega)$, $T^k(\omega) \leftarrow T^{k-1}(\omega)$, $T^k_{B_1}(\omega) \leftarrow T^{k-1}_{B_1}(\omega)$, $r^k(\omega) \leftarrow r^{k-1}(\omega)$;

  Let $y^k(\omega) \leftarrow y^{k-1}(\omega)$;

endfor

if $y^k(\omega) \notin \mathbb{Z} \times \mathbb{Z}^{n_2}_+$ *for some* $\omega \in \Omega$ **then**

  for $\omega \in \Omega$ *with* $y^k(\omega) \notin \mathbb{Z} \times \mathbb{Z}^{n_2}_+$ **do**

    Let $i^* = i^*(\omega) := \min\{i : i \in \{0, \dots, n_2\}, y^k_i(\omega) \notin \mathbb{Z}\}$;

    Let $W^k_{B_2}(\omega)$ be the basis matrix corresponding to $y^k(\omega)$;

    Let $d^k_{i^*}(\omega) := \left[-(W^k_{B_2}(\omega))^{-1}_{i^*} T^k_{B_1}(\omega) {A^k_{B_1}}^{-1} \quad {W^k_{B_2}(\omega)}^{-1}_{i^*} \right]$, where $(W^k_{B_2}(\omega))^{-1}_{i^*}$ is the i^*th row of the matrix ${W^k_{B_2}(\omega)}^{-1}$;

    Generate a Gomory cut from the source row $d^k_{i^*}(\omega) \begin{bmatrix} A^k & \mathbf{0} \\ T^k(\omega) & W^k(\omega) \end{bmatrix} \begin{bmatrix} x \\ y(\omega) \end{bmatrix} = d^k_{i^*}(\omega) \begin{bmatrix} b^k \\ r^{k-1}(\omega) \end{bmatrix}$;

    Add the Gomory cut to $\mathrm{SP}^k(x, \omega)$, update $W^k(\omega), T^k(\omega), r^k(\omega)$;

    Solve $\mathrm{SP}^k(x^k, \omega)$ using the lexicographic dual simplex method to obtain $y^k(\omega)$;

  endfor

end

if $x^k \in \mathbb{Z}^{n_1+2}_+$ *and* $y^k(\omega) \in \mathbb{Z} \times \mathbb{Z}^{n_2}_+$ *for all* $\omega \in \Omega$ **then**

  $UB \leftarrow \min\{UB, Q(\bar{c}^\top \bar{x}^k + \sum_{\omega \in \Omega} p_\omega f^k(\bar{x}^k, \omega))\}$;

end

Return optimality cut (2.16) and $y^k(\omega), UB, W^k(\omega), T^k(\omega), r^k(\omega)$ for all $\omega \in \Omega$ to Algorithm 1;

Iteration 2. $k = 2$. Because $UB - LB > 0$, we solve MP^1, with lexicographic dual simplex method and obtain $x^1 = (0, 5, 0, 331)$. Then we generate master problem MP^2, in which $A^2 = A^1$ and $b^2 = b^1$. We update the lower bound as $LB = -571$. Let $x^2 = x^1$.

We solve subproblems $SP^1(x^2, \omega_i)$, $i \in [1, 3]$, with lexicographic simplex method to obtain $y^1(\omega_i)$. Then we generate subproblem $SP^2(x, \omega_i)$ for $i \in [1, 3]$, in which $W^2(\omega_i) = W^1(\omega_i)$, $T^2(\omega_i) = T^1(\omega_i)$, and $r^2(\omega_i) = r^1(\omega_i)$. Let $y^2(\omega_i) = y^1(\omega_i)$. We have $y^2(\omega_1) = (123.43, 0, 4.57, 0, 1.14) \notin \mathbb{Z}^5$. We add a parametric Gomory cut

$$6y_1(\omega_1) + 3y_2(\omega_1) + 5y_3(\omega_1) + 5y_4(\omega_1) \leq 34 - x_1 - 3x_2 \tag{2.21}$$

to $SP^2(x, \omega_1)$, re-solve $SP^2(x^2, \omega_1)$ and obtain $y^2(\omega_1) = (122.4, 0, 5, 0, 0.8)$. In addition, $y^2(\omega_2) = (171, 0, 9, 0, 0)$, and $y^2(\omega_3) = (28, 0, 0, 0, 1)$, which are both integral. But we do not update the upper bound because $y^2(\omega_1) \notin \mathbb{Z}^5$. We update the master *problem* MP^2 *with the optimality cut*

$$417x_1 + 678x_2 - 15x_3 + 15x_4 \leq 8211. \tag{2.22}$$

Let $l_2 = l_1 + 1 = 5$*.*

Iteration 3. $k = 3$*. Because* $UB - LB > 0$*, we solve* MP^2 *with lexicographic dual simplex method and obtain the optimal solution* x^2 *as* $(0, 5, 0, 321.4)$*. Then we generate master problem* MP^3*, in which* $A^3 = A^2$ *and* $b^3 = b^2$*. Let* $x^3 = x^2$*. Because* $x^3 \notin \mathbb{Z}^4_+$*, we add a Gomory cut* $27x_1 + 46x_2 - x_3 + x_4 \leq 551$ *to the master problem* MP^3 *and re-solve it using lexicographic dual simplex to obtain* $x^3 = (0, 5, 0, 321) \in \mathbb{Z}^4_+$*. We update the lower bound as* $LB = -561$*.*

With $x^3 = (0, 5, 0, 321)$*, we solve the subproblems* $SP^2(x^3, \omega_i)$*,* $i \in [1, 3]$ *with lexicographic simplex method to obtain* $y^2(\omega_i)$*. Then we generate subproblem* $SP^3(x, \omega_i)$ *for* $i \in [1, 3]$*, in which* $W^3(\omega_i) = W^2(\omega_i)$*,* $T^3(\omega_i) = T^2(\omega_i)$*, and* $r^3(\omega_i) = r^2(\omega_i)$*. Let* $y^3(\omega_i) = y^2(\omega_i)$*. We have* $y^3(\omega_1) = (122.4, 0, 5, 0, 0.8) \notin \mathbb{Z}^5$*. We add a parametric Gomory cut*

$$3y_1(\omega_1) + 2y_2(\omega_1) + 3y_3(\omega_1) + 3y_4(\omega_1) \leq 22 - 2x_2 \tag{2.23}$$

to $SP^3(x, \omega_1)$*, re-solve* $SP^3(x^3, \omega_1)$ *and obtain* $y^3(\omega_1) = (120, 0, 6, 0, 0)$*. In addition,* $y^3(\omega_2) = (171, 0, 9, 0, 0) \in \mathbb{Z}^5$ *and* $y^3(\omega_3) = (28, 0, 0, 0, 1) \in \mathbb{Z}^5$*. Because* $x^3 \in \mathbb{Z}^4$*, and* $y^3(\omega_i) \in \mathbb{Z}^5$*,* $i \in [1, 3]$*, we update the upper bound as* $UB = -559$*. We also update the master problem* MP^3 *with the optimality cut*

$$69x_1 + 150x_2 - 3x_3 + 3x_4 \leq 1707. \tag{2.24}$$

Let $l_3 = l_2 + 2 = 7$*.*

Iteration 4. $k = 4$*. Because* $UB - LB > 0$*, we solve* MP^3 *with lexicographic simplex method and obtain the optimal solution* $x^3 = (0, 4.5, 0, 344) \notin \mathbb{Z}^4_+$*. Let* $x^4 = x^3$*. Then we generate master problem* MP^4 *as the same as* MP^3*, and add two Gomory cuts* $26x_1 + 47x_2 - x_3 + x_4 \leq 555$ *and* $25x_1 + 48x_2 - x_3 + x_4 \leq 559$ *to* MP^4*. After re-solving it with lexicographic dual simplex, we obtain an integer solution* $x^4 = (0, 5, 0, 319)$*. The lower bound is updated as* $LB = -559$*.*

With $x^4 = (0, 5, 0, 319)$*, we solve the subproblems* $SP^3(x^4, \omega_i)$*,* $i \in [1, 3]$ *to obtain* $y^3(\omega_i)$*. Let* $y^4(\omega_i) = y^3(\omega_i)$*. The solutions* $y^4(\omega_1) = (120, 0, 6, 0, 0) \in \mathbb{Z}^5$*,* $y^4(\omega_2) = (171, 0, 9, 0, 0) \in \mathbb{Z}^5$ *and* $y^4(\omega_3) = (28, 0, 0, 0, 1) \in \mathbb{Z}^5$*. Hence, we get the upper bound* $UB = -559$*, which is equal to the current lower bound. Therefore, we have found the optimal integer solution* $\bar{x} = (0, 5)$ *and* $y = ((120, 0, 6, 0, 0), (171, 0, 9, 0, 0), (28, 0, 0, 0, 1))$*.*

3. Finite Convergence. Next, we establish the convergence of the proposed algorithm.

THEOREM 3.1. *Suppose that assumptions (A1)-(A6) are satisfied, then Algorithm 1 finds an optimal solution to* (1.1)-(1.6) *in finitely many iterations.*

Proof. In iteration k, where $k \equiv 0 \ mod \ q$, we solve the master problem MP^k with a Gomory cutting plane algorithm in a finite number of steps to obtain $x^k \in \mathbb{Z}_+^{n_1+2}$, and solve subproblems $\text{SP}^k(x^k, \omega)$ to obtain $y^k(\omega)$ for each $\omega \in \Omega$. In iteration $k+1$, there are three cases to consider.

(i) Suppose that $y^k(\omega) \in \mathbb{Z} \times \mathbb{Z}_+^{n_2}$ for all $\omega \in \Omega$, and $x^{k+1} = x^k$. Then the solution $(\bar{x}^k, \{y^k(\omega)\}_{\omega \in \Omega})$ must be the optimal solution to (1.1)-(1.6) because $x^{k+1}_{n_1+1} - x^{k+1}_{n_1+2} \geq \sum_{\omega \in \Omega} Q p_\omega f^k(\bar{x}^{k+1}, \omega)$, which is a consequence of the optimality cut (2.16) that is added at iteration k. Thus, $LB = Q\bar{c}^\top \bar{x}^{k+1} + x^{k+1}_{n_1+1} - x^{k+1}_{n_1+2} \geq Q(\bar{c}^\top \bar{x}^k + \sum_{\omega \in \Omega} p_\omega f^k(\bar{x}^{k+1}, \omega)) = UB$. Therefore, $LB = UB$, and we found the optimal solution.

(ii) Suppose that $x^{k+1} \neq x^k$, then x^k must violate the optimality cut generated at the end of iteration k. Therefore, x^k will not be visited again in a future iteration.

(iii) Suppose that $y^k(\omega) \notin \mathbb{Z} \times \mathbb{Z}_+^{n_2}$ for some $\omega \in \Omega$, and $x^{k+1} = x^k$. Note that for a given $(\bar{x}, \omega) \in \bar{X} \times \Omega$, because of Assumptions (A3), (A4) and (A6), there exists an integer $K(\bar{x}, \omega) < +\infty$ such that the optimal solution $y(\omega) \in Y(\bar{x}, \omega)$ can be found by Algorithm 2 in at most $K(\bar{x}, \omega)$ iterations. This is essentially because of the finite convergence of the Gomory cutting plane method using lexicographic dual simplex [10]. (We also refer the reader to Gade et al. [9] for the proof of the finite convergence of Gomory cutting plane method in stochastic context.) Therefore, it takes at most $\max_{\omega \in \Omega} K(\bar{x}^k, \omega)$ iterations to add parametric Gomory cuts to the second-stage subproblems to obtain an integral second-stage solution. Let $\ell \in [1, \max_{\omega \in \Omega} K(\bar{x}^k, \omega)]$ be the largest number such that $x^{k+1}, \ldots, x^{k+\ell}$ are all equal to x^k, and $y^{k+1}(\omega), \ldots, y^{k+\ell}(\omega)$ are all fractional. Then in iteration $k + \ell + 1$,

- if $x^{k+\ell+1} = x^k$ and $y^{k+\ell+1}(\omega) \in \mathbb{Z} \times \mathbb{Z}_+^{n_2}, \omega \in \Omega$, then we have found an optimal solution (the same argument as in case (i));
- if $x^{k+\ell+1} \neq x^k$, then x^k will not be visited again in a future iteration (the same argument as in case (ii));

Hence, any $\bar{x} \in \bar{X}$ will be visited in at most a finite number of consecutive iterations. In every q-th iteration, an integral first-stage solution is found by Algorithm 1 with finitely many Gomory cuts [10] using lexicographic dual simplex for each LP. From Assumption (A6) there are only finitely many $\bar{x} \in \bar{X}$ that can be visited. Also, Ω is finite as stated in Assumption (A5). Hence, in the worst case, there exists an integer $K = \sum_{\bar{x} \in \bar{X}} \max_{\omega \in \Omega} K(\bar{x}, \omega)$ such that either Algorithm 1 terminates in $k < qK$ iterations or $y^k(\omega) \in \mathbb{Z} \times \mathbb{Z}_+^{n_2}$ and $f^{k-1}(\bar{x}^k, \omega) = f(\bar{x}^k, \omega)$ for all $k \geq qK$. Then the convergence of Algorithm 1 follows from the convergence of the Benders' decomposition method. ∎

4. A Branch-and-Cut Based Decomposition Algorithm. Exploiting the finite convergence of the branch-and-bound [13] and branch-and-cut methods, we develop an alternative branch-and-cut based decomposition algorithm. For ease of exposition, we first describe the branch-and-bound (B&B) implementation with a breadth-first strategy. Let FP^t denote the master problem at the t-th node of the B&B tree. This problem is the same as that defined in Section 2.2, but in which constraints (2.15) include the bounds on the variables introduced during the B&B

process, and constraints (2.16) only include the optimality cuts generated that are valid for node t of the branch-and-bound tree. Let $\mathrm{FP}^0 = \mathrm{MP}^0$. Let $\mathcal{L}$ be a collection of FP^t problems for all leaf nodes, t, in the B&B process, and LB^t and UB^t be the corresponding lower and upper bounds, respectively. Initially, $\mathcal{L}$ contains problem FP^0 with upper bound $UB^0 = +\infty$ and lower bound $LB^0 = -\infty$. Let T^* denote the objective function value of the incumbent solution. Initially, $T^* = +\infty$. In addition, we denote $\mathrm{CP}^t(x, \omega)$ as the subproblems defined in Section 2.2, but in which constraint (2.17) only includes the parametric Gomory cuts generated that are valid for node t of the branch-and-bound tree. In iteration k, if list $\mathcal{L} \neq \emptyset$, then let $j(k)$ be the smallest index among the problems in the list $\mathcal{L}$. Also we denote the optimal solutions to $\mathrm{FP}^{j(k)}$ and $\mathrm{CP}^{j(k)}(x, \omega)$ as $x^{j(k)}$ and $y^{j(k)}(\omega)$ for $\omega \in \Omega$, respectively. We solve problem $\mathrm{FP}^{j(k)}$ by lexicographic dual simplex method to obtain $x^{j(k)}$. Let $LB^{j(k)} = c^\top x^{j(k)}$.

- If the lower bound $LB^{j(k)} \geq T^*$, then we prune node $j(k)$ because it is impossible to obtain a better integer solution branching from this node.
- If the lower bound $LB^{j(k)} < T^*$ and $(x_1^{j(k)}, \ldots, x_{n_1}^{j(k)})$ is fractional, then we branch on the first fractional component of $\bar{x}^{j(k)}$ at node $j(k)$ in the B&B tree to obtain the problems FP^{2k-1} and FP^{2k}. Let problems FP^{2k-1} and FP^{2k} substitute problem $\mathrm{FP}^{j(k)}$ in the list $\mathcal{L}$. Also let the second-stage problems $\mathrm{CP}^{2k-1}(x, \omega)$ and $\mathrm{CP}^{2k}(x, \omega)$ be the same as $\mathrm{CP}^{j(k)}(x, \omega)$, and $UB^{2k-1} = UB^{2k} = +\infty$.
- If the lower bound $LB^{j(k)} < T^*$ and $(x_1^{j(k)}, \ldots, x_{n_1}^{j(k)})$ is integral, then we solve problem $\mathrm{CP}^{j(k)}(x^{j(k)}, \omega)$ for all $\omega \in \Omega$ by lexicographic dual simplex method. For each $\omega \in \Omega$, if $y^{j(k)}(\omega)$ is fractional, then we add a parametric Gomory cut to problem $\mathrm{CP}^{j(k)}(x^{j(k)}, \omega)$ and re-solve $\mathrm{CP}^{j(k)}(x^{j(k)}, \omega)$ by lexicographic dual simplex method to obtain a new solution $y^{j(k)}(\omega)$. If $y^{j(k)}(\omega) \in \mathbb{Z} \times \mathbb{Z}_+^{n_2}$ for all $\omega \in \Omega$, then we update $UB^{j(k)}$ by $\min\{UB^{j(k)}, Q(\bar{c}^\top \bar{x}^{j(k)} + \sum_{\omega \in \Omega} p_\omega y_0^{j(k)}(\omega))\}$.
  - If $UB^{j(k)} - LB^{j(k)} \leq \epsilon$ and $UB^{j(k)} < T^*$, then we let $T^* = UB^{j(k)}$ and update incumbent solution $(\bar{x}^*, \{y^*(\omega)\}_{\omega \in \Omega})$ by $(\bar{x}^{j(k)}, \{y^{j(k)}(\omega)\}_{\omega \in \Omega})$.
  - If $UB^{j(k)} - LB^{j(k)} \leq \epsilon$ and $UB^{j(k)} \geq T^*$, then we prune node $j(k)$ because it does not improve the objective function value.
  - If $UB^{j(k)} - LB^{j(k)} > \epsilon$, then we add the optimality cut obtained from $\mathrm{CP}^{j(k)}(x^{j(k)}, \omega)$ to the problem $\mathrm{FP}^{j(k)}$.

Increment iteration index k, and repeat until $\mathcal{L} \neq \emptyset$.

Once the list $\mathcal{L}$ becomes empty, we output the incumbent solution $(\bar{x}^*, \{y^*(\omega)\}_{\omega \in \Omega})$ as the optimal solution and T^* as the optimal objective function value. The detailed algorithm is described by Algorithm 3. Note that we can have an alternative implementation, where we add any valid inequalities (Gomory or any other class) when solving the master problem, which we refer to as the branch-and-cut based decomposition algorithm.

Algorithm 3: Branch-and-bound based algorithm for solving SIP

$\mathrm{FP}^0 \leftarrow \mathrm{MP}^0$, $\mathcal{L} \leftarrow \{\mathrm{FP}^0\}$, $k \leftarrow 1$, $T^* \leftarrow +\infty$, $UB^0 \leftarrow +\infty$, $LB^0 \leftarrow -\infty$, $\mathrm{CP}^0(x,\omega) \leftarrow \mathrm{SP}^0(x,\omega)$ for $\omega \in \Omega$;
while $\mathcal{L} \neq \emptyset$ **do**
  $j(k) \leftarrow \min\{j : \mathrm{FP}^j \in \mathcal{L}\}$;
  Solve $\mathrm{FP}^{j(k)}$ by lexicographic simplex method to obtain $x^{j(k)}$;
  $LB^{j(k)} \leftarrow c^\top x^{j(k)}$;
  if $LB^{j(k)} \geq T^*$ **then**
    $\mathcal{L} \leftarrow \mathcal{L} \setminus \{\mathrm{FP}^{j(k)}\}$;
  else if $LB^{j(k)} < T^*$ *and* $\bar{x}^{j(k)} \notin \mathbb{Z}_+^{n_1}$ **then**
    Branch on $x_{i(k)}^{j(k)}$ with $i(k) = \min\{i : x_i^{j(k)} \notin \mathbb{Z}_+, i \in [1, n_1]\}$ to obtain FP^{2k-1}, FP^{2k};
    For $\omega \in \Omega$, $\mathrm{CP}^{2k-1}(x,\omega) \leftarrow \mathrm{CP}^{j(k)}(x,\omega), \mathrm{CP}^{2k}(x,\omega) \leftarrow \mathrm{CP}^{j(k)}(x,\omega)$;
    $UB^{2k-1} \leftarrow +\infty$, $UB^{2k} \leftarrow +\infty$;
    $\mathcal{L} \leftarrow \mathcal{L} \cup \{\mathrm{FP}^{2k-1}, \mathrm{FP}^{2k}\} \setminus \{\mathrm{FP}^{j(k)}\}$;
  else if $LB^{j(k)} < T^*$ *and* $\bar{x}^{j(k)} \in \mathbb{Z}_+^{n_1}$ **then**
    for $\omega \in \Omega$ **do**
      Solve $\mathrm{CP}^{j(k)}(x^{j(k)}, \omega)$ by lexicographic simplex method to obtain $y^{j(k)}(\omega)$;
      if $y^{j(k)}(\omega) \notin \mathbb{Z} \times \mathbb{Z}_+^{n_2}$ **then**
        Generate a parametric Gomory cut for $y_{u(k)}^{j(k)}(\omega)$ with $u(k) = \min\{u : y_u^{j(k)}(\omega) \notin \mathbb{Z}, u \in [0, n_2]\}$;
        Add the parametric Gomory cut to $\mathrm{CP}^{j(k)}(x,\omega)$, and solve $\mathrm{CP}^{j(k)}(x^{j(k)},\omega)$ by lexicographic dual simplex method to obtain a new $y^{j(k)}(\omega)$;
      end
    endfor
    if $y^{j(k)}(\omega) \in \mathbb{Z} \times \mathbb{Z}_+^{n_2}$ *for all* $\omega \in \Omega$ **then**
      $UB^{j(k)} \leftarrow \min\{UB^{j(k)}, Q(\bar{c}^\top \bar{x}^{j(k)} + \sum_{\omega \in \Omega} p_\omega y_0^{j(k)}(\omega))\}$;
    end
    if $UB^{j(k)} - LB^{j(k)} \leq \epsilon$ *and* $UB^{j(k)} < T^*$ **then**
      $(\bar{x}^*, \{y^*(\omega)\}_{\omega \in \Omega}) \leftarrow (\bar{x}^{j(k)}, \{y^{j(k)}(\omega)\}_{\omega \in \Omega})$, $T^* \leftarrow UB^{j(k)}$, $\mathcal{L} \leftarrow \mathcal{L} \setminus \{FP^{j(k)}\}$;
    else if $UB^{j(k)} - LB^{j(k)} \leq \epsilon$ *and* $UB^{j(k)} \geq T^*$ **then**
      $\mathcal{L} \leftarrow \mathcal{L} \setminus \{FP^{j(k)}\}$;
    else if $UB^{j(k)} - LB^{j(k)} > \epsilon$ **then**
      Add optimality cut derived from $\mathrm{CP}^{j(k)}(x^{j(k)},\omega)$ to $\mathrm{FP}^{j(k)}$;
  $k \leftarrow k + 1$;
end
Return $(\bar{x}^*, \{y^*(\omega)\}_{\omega \in \Omega})$ and T^*;

PROPOSITION 4.1. *Suppose that assumptions (A1)-(A6) are satisfied, then Algorithm 3 finds an optimal solution to* (1.1)-(1.6) *in finitely many iterations.*

Proof. The finite convergence to an integer first-stage solution follows from the finiteness of the branch-and-bound process. Suppose that the solution $\bar{x}^j$ to problem FP^j is integral, it takes finitely many iterations to find the optimal solution $y^j(\omega) \in Y(\bar{x}^j, \omega)$ for each $\omega \in \Omega$ because of the finite convergence of Gomory cutting plane algorithm. In addition, at each node j, it takes finitely many iterations to obtain the optimal solutions x^j and $y^j(\omega)$ because of the finite convergence of Benders' decomposition method [3]. Therefore, Algorithm 3 finds an optimal solution to (1.1)-(1.6) in finitely many iterations. □

Note that Proposition 4.1 also holds for the branch-and-cut based decomposition algorithm using any valid inequalities for the master problem. Next we illustrate Algorithm 3 on Example 1.

Example 1. (Continued). Algorithm 3 performs the same iterations as the first three iterations of Algorithm 1 until it obtains a first-stage solution $x = (0, 4.5, 0, 344)$. *(Note that we only branch on the variables* $\bar{x}$ *in Algorithm 3.) Thus, at the root node in Algorithm 3, we have* $x^0 = (0, 4.5, 0, 344)$. *Also,*

$$FP^0 = \min\{-18x_1 - 48x_2 + x_3 - x_4 : (2.3), (2.4), (2.20), (2.22), (2.24)\},$$

$y^0(\omega_1) = (120, 0, 6, 0, 0)$, $y^0(\omega_2) = (171, 0, 9, 0, 0)$, $y^0(\omega_3) = (28, 0, 0, 0, 1)$,

$$CP^0(x, \omega_1) = \min\{-y_0(\omega_1) : (2.5), (2.6), (2.7), (2.18), (2.21), (2.23)\},$$

$$CP^0(x, \omega_2) = \min\{-y_0(\omega_2) : (2.8), (2.9), (2.10)\},$$

$$CP^0(x, \omega_3) = \min\{-y_0(\omega_3) : (2.11), (2.12), (2.13), (2.19)\},$$

and $\mathcal{L} = \{FP^0\}$. *Note that for Algorithm 3 the superscripts are the indices of the B&B nodes. We demonstrate Algorithm 3 starting with* $k = 4$, *where* $j(k) = 0$.

Because $x_2^0 \notin \mathbb{Z}_+$, *we branch on the root node to generate two leaf nodes with two bounding inequalities* $x_2 \geq 5$ *and* $x_2 \leq 4$. *Therefore,*

$$FP^1 = \min\{-18x_1 - 48x_2 + x_3 - x_4 : (2.3), (2.4), (2.20), (2.22), (2.24), x_2 \geq 5\},$$

and

$$FP^2 = \min\{-18x_1 - 48x_2 + x_3 - x_4 : (2.3), (2.4), (2.20), (2.22), (2.24), x_2 \leq 4\},$$

In addition, $CP^1(x, \omega_i) = CP^2(x, \omega_i) = CP^0(x, \omega_i)$, $i \in [1, 3]$, *and* $UB^1 = UB^2 = +\infty$. *Note that* $\mathcal{L} = \{FP^1, FP^2\}$.

Iteration 5. $k = 5$. *Because* $j(k) = 1$, *we solve problem* FP^1 *by lexicographic dual simplex method, we get* $x^1 = (0, 5, 0, 319)$ *and* $LB^1 = -559$, *Because* $LB^1 < T^* = +\infty$ *and* $(x_1^1, x_2^1) = (0, 5) \in \mathbb{Z}_+^2$, *we solve* $CP^1(\omega)$ *for each* $\omega \in \Omega$, *and obtain* $y^1(\omega_1) = (120, 0, 6, 0, 0) \in \mathbb{Z}^5$, $y^1(\omega_2) = (171, 0, 9, 0, 0) \in \mathbb{Z}^5$, *and* $y^1(\omega_3) = (28, 0, 0, 0, 1) \in \mathbb{Z}^5$. *Then we update the upper bound as* $UB^1 = -559$. *Because* $LB^1 = UB^1 = -559$,

$(\bar{x}^1, \{y^1(\omega)\}_{\omega\in\Omega})$ is an incumbent solution and $T^* = -559$. Note that $\mathcal{L} = \{FP^2\}$.

Iteration 6. $k = 6$. Because $j(k) = 2$, we solve problem FP^2 by lexicographic dual simplex method, we get $x^2 = (0, 4, 0, 366.6)$ and $LB^2 = -558.6$. Because $LB^2 > T^* = -559$, node 2 is pruned. Then $\mathcal{L} = \emptyset$.

Thus, the incumbent solution $\bar{x} = (0, 5)$, $y(\omega_1) = (120, 0, 6, 0, 0)$, $y(\omega_2) = (171, 0, 9, 0, 0)$, $y(\omega_3) = (28, 0, 0, 0, 1)$ is optimal. Compared with Algorithm 1, Algorithm 3 solves this instance with two fewer Gomory cuts added to the master, by instead branching on a fractional variable.

5. Preliminary Computational Results. To demonstrate the performance of the proposed decomposition algorithms, we test them on various instances of different sizes. We consider $m \in \{50, 100, 200, 250\}$, $n_1 = 5$, $n_2 \in \{5, 10\}$, $a \in \{5, 10\}$, $t(\omega) \in \{5, 10\}$. We assume that each scenario is equally likely, hence $Q = m$. For each setting $m.n_2.a.t(\omega)$, five random instances are generated, and we report the average performance. Following the data generation in Hemmecke and Schultz [11] and Ahmed et al. [1], the first-stage cost function $\bar{c}$, first-stage matrix $\bar{A}$, first-stage right-hand side $\bar{b}$, second-stage cost function $g(\tilde{\omega})$, technology matrix $\bar{T}(\tilde{\omega})$, recourse matrix $W(\tilde{\omega})$, and right-hand-side vector $r(\tilde{\omega})$ follow discrete distributions on intervals:

- $\bar{c}_j \in [-6, -1]$, $j = 1, \ldots, n_1$.
- $\bar{A}_{ij} \in [0, 1]$, $i = 1, \ldots, a$, $j = 1, \ldots, n_1$.
- $\bar{b}_i \in [5, 10]$, $i = 1, \ldots, a$.
- $g_j(\tilde{\omega}) \in [-40, -20]$, $j = 1, \ldots, n_2$.
- $\bar{T}_{ij}(\tilde{\omega}) \in [0, 1]$, $i = 1, \ldots, t(\omega)$, $j = 1, \ldots, n_1$.
- $W_{ij}(\tilde{\omega}) \in [10, 20]$, $i = 2, \ldots, t(\omega)$, $j = 1, \ldots, n_2$.
- $\bar{r}_i(\tilde{\omega}) \in [50, 350]$, $i = 1, \ldots, t(\omega)$.

We solve our instances by branch-and-cut based decomposition Algorithm 3 (denoted by BCDG). In algorithm BCDG, we allow to add at most 10 Gomory cuts to the problem FP at each node when $\bar{x} \notin \mathbb{Z}_+^{n_1}$ before branching on the fractional component. For a particular $x \in \mathbb{Z}_+^{n_1+2}$, if there still exists $i \in [1, m]$ such that $y(\omega_i) \notin \mathbb{Z}_+^{n_2}$ after 10 consecutive iterations, then we call IBM ILOG CPLEX to solve the subproblems as IPs to find an integer solution $\{y(\omega)\}_{\omega\in\Omega}$ to update the upper bound. Because of the limited flexibility of customizing the solution process in a commercial optimization software, such as CPLEX, we implement lexicographic dual simplex method, branch-and-bound process, Gomory cut generation on our own (with C++ language) instead of calling any external solvers. The only time we use CPLEX is to obtain upper bounds. We run our codes on a 3.40-GHz Intel(R) Core(TM) i7-3770 processor with 8 GB RAM. For sake of comparison, we also solve the DEF of these instances by IBM ILOG CPLEX 12.5 with default CPLEX setting without preprocessing (denoted by CPLEX). A time limit of 1 hour is imposed.

In Table 5.1, we summarize the performance of algorithms BCDG and CPLEX with different settings of $m.n_2.a.t(\omega)$. Column **D. itrtn** reports the number of iterations that Algorithm BCDG takes. Column **S. itrtn** reports the number of times the lexicographic dual simplex method is called. Column **Cuts** reports the number of Gomory cuts added. For Algorithm BCDG, the first and second numbers in the parenthesis are the numbers of Gomory cuts added to the master problem and subproblems, respectively. Column **B-B nodes** reports the number of B&B tree nodes explored. Column **Gap** shows the gap between the best lower bound and the optimal objective function value, and column **Time** reports the solution time in seconds. In addition, in column **CPLEX#**, we report the number of instances for which CPLEX is called in Algorithm BCDG for solving the subproblems to integer optimality to obtain an upper bound. In column **Int-x(imp)**, we report the number of integer first-stage solutions found by BCDG before the optimal solution is obtained. The number in parenthesis in this column is the number of integer solutions that lead to improved upper bounds.

TABLE 5.1
Computational results on test instances

$m.n_2.a.t(\omega)$	Algorithm	Time	D.itrtn	S.itrtn	B-B nodes	Gap(%)	CPLEX#	Cuts	Int-x(imp)
50.5.5.5	BCDG	9.1	62.5	2972.8	147.8	0	2	(729.8, 59.6)	12.4(8)
	CPLEX	≥3600	-	-	929971.2	0.02	-	128.4	-
50.5.10.10	BCDG	27.8	111.4	1235.2	157.8	0	2	(967, 68.4)	8.5(3.6)
	CPLEX	≥3600	-	-	748274	0.32	-	134.6	-
50.10.5.10	BCDG	26.8	115.4	2703.2	157.4	0	3	(158.4, 248.4)	5.8(1.8)
	CPLEX	≥3600	-	-	873901.2	0.28	-	128.4	-
100.5.5.5	BCDG	11.8	153.4	2125.6	593	0	3	(262, 599.2)	14.4(5)
	CPLEX	≥3600	-	-	3423452.2	0.32	-	321.2	-
100.5.10.10	BCDG	21.1	200.4	2266.2	118.2	0	4	(566, 897.4)	22.2(7.4)
	CPLEX	≥3600	-	-	7321943.4	0.35	-	439.2	-
100.10.5.10	BCDG	88.5	108.2	2213.2	232.6	0	4	(1421.2, 698.6)	17.6(5.6)
	CPLEX	≥3600	-	-	5615818.2	0.28	-	283.4	-
200.5.5.5	BCDG	114.1	106.6	1510.2	203.8	0	5	(1102.2, 425.4)	15.4(3.6)
	CPLEX	≥3600	-	-	2942659.2	0.32	-	358.4	-
200.5.10.10	BCDG	208.2	206.4	2470.6	379.8	0	5	(2943.4, 212.4)	22.4(7.6)
	CPLEX	≥3600	-	-	3218475.4	0.45	-	451.2	-
200.10.5.10	BCDG	111.1	155	1448	234.6	0	5	(454.6, 921.6)	23(11.4)
	CPLEX	≥3600	-	-	3141551.4	0.25	-	512.6	-
250.5.5.5	BCDG	137.4	210.4	2815.6	303	0	5	(1377, 598.2)	22(9.4)
	CPLEX	≥3600	-	-	3223532.6	0.41	-	657.2	-
250.5.10.10	BCDG	127.6	198.2	2134.2	144.2	0	5	(806, 493.4)	32.2(11.6)
	CPLEX	≥3600	-	-	2582267.2	0.66	-	544.2	-
250.10.5.10	BCDG	121.4	279.2	2729.4	318.6	0	5	(1555.8, 457.4)	22.2(9.4)
	CPLEX	≥3600	-	-	4780321.2	0.39	-	952.4	-

Comparing algorithm BCDG with CPLEX, we see that despite the disadvantage of not having utilized the state-of-the-art linear programming solver of CPLEX, our in-house implementation of BCDG already outperforms the branch-and-cut method for the DEF employed by CPLEX for problems with modest number of scenarios, decision variables and constraints. BCDG solves all of these instances within a few minutes, whereas CPLEX is not able to solve any of them in an hour. CPLEX explores millions of B&B tree nodes and essentially resorts to enumeration. In contrast, BCDG effectively uses the bound information from the second-stage value function approximation to solve the problem exploring a few hundred B&B tree nodes. We also see from the last column that less than half of the integer feasible solutions to the master problem give improved upper bounds, justifying the approximation of the second-stage value function instead of calculating it exactly. We can conclude from our preliminary experience that the decomposition method significantly reduces the computational burden, and the branch-and-bound and Gomory cuts help expedite the convergence to an optimal solution for these instances. We also tested Algorithm 1 on this class of instances. However, we observed that Algorithm 1 is forced to stop before the time limit is reached for instances with more than five scenarios, because of the computer memory limit and numerical issues. Therefore, we do not report our results with this algorithm.

6. Conclusion. We study a class of two-stage stochastic pure integer programs with general integer variables in both stages (SIP). We consider a very general class of problems, where the cost function of the second-stage decision variables, technology and recourse matrices, and the right-hand-side of the constraints could be affected by random parameters. We assume that the random parameters have finite support. Instead of solving the large-size deterministic equivalent of the two-stage SIP, we propose a decomposition algorithm based on Benders' method to solve the second-stage problem for each scenario separately, and return an approximation of the second-stage cost function to the first-stage problem. Our method generates Gomory cuts parameterized with respect to the first-stage decision variables, i.e., they are valid for the deterministic equivalent. We also propose an alternative algorithm that implements Benders' decomposition method in the branch-and-bound process. We prove that the optimal solution can be found within finitely many iterations. Our results with a preliminary implementation of our algorithm are very encouraging. As part of our future work, we plan to develop a more robust implementation of our algorithms to solve SIPs of larger sizes.

Acknowledgments. We would like to thank two anonymous referees whose comments improved an earlier version of this paper. We also acknowledge Suvrajeet Sen for the beneficial discussions.

REFERENCES

[1] S. Ahmed, M. Tawarmalani, and N. V. Sahinidis, *A finite branch-and-bound algorithm for two-stage stochastic integer programs*, Mathematical Programming, 100 (2004), pp. 355–377.

[2] E. Balas, S. Ceria, and G. Cornuéjols, *A lift-and-project cutting plane algorithm for mixed 0-1 programs*, Mathematical Programming, 58 (1993), pp. 295–324.

[3] J. Benders, *Partitioning procedures for solving mixed-variables programming problems*, Numerische Mathematik, 4 (1962), pp. 238–252.

[4] C. E. Blair and R. G. Jeroslow, *The value function of an integer program*, Mathematical Programming, 23 (1982), pp. 237–273.

[5] C. C. Carøe and R. Schultz, *Dual decompostion in stochastic integer programming*, Operations Research Letters, 24 (1999), pp. 37–45.

[6] C. C. Carøe and J. Tind, *A cutting-plane approach to mixed 0-1 stochastic integer programs*, European Journal of Operational Research, 101 (1997), pp. 306–316.

[7] ———, *L-shaped decomposition of two-stage stochastic programs with integer recouse*, Mathematical Programming, 101 (1998), pp. 306–316.

[8] D. Gade and S. Küçükyavuz, *Pure cutting-plane algorithms and their convergence*, in Wiley Encyclopedia of Operations Research and Management Science, J. J. Cochran, L. A. Cox,

P. Keskinocak, J. P. Kharoufeh, and J. C. Smith, eds., John Wiley & Sons, Inc., 2013, pp. 1–11.

[9] D. Gade, S. Küçükyavuz, and S. Sen, *Decomposition algorithms with parametric Gomory cuts for two-stage stochastic integer programs*, Mathematical Programming, 144 (2014), pp. 39–64.

[10] R. E. Gomory, *Outline of an algorithm for integer solutions to linear programs*, Bulletin of the American Mathematical Society, 64 (1958), pp. 275–278.

[11] R. Hemmecke and R. Schultz, *Decomposition of test sets in stochastic integer programming*, Mathematical Programming, 94 (2000), pp. 323–341.

[12] N. Kong, A. J. Schaefer, and B. Hunsaker, *Two-stage integer programs with stochastic right-hand sides: a superadditive dual formulation*, Mathematical Programming, 108 (2006), pp. 275–296.

[13] A. Land and A. Doig, *An automatic method of solving discrete programming problems*, Econometrica: Journal of the Econometric Society, (1960).

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

[15] C. P. M. Lubin, K. Martin and B. Sandikci, *On parallelizing dual decomposition in stochastic integer programming*, Operations Research Letters, 41 (2013), pp. 252–258.

[16] C. H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*, Dover Publications, 1998.

[17] R. Schultz, L. Stougie, and M. H. van der Vlerk, *Solving stochastic programs with integer recourse by enumeration: A framework using Gröbner basis*, Mathematical Programming, 183 (1998), pp. 229–252.

[18] S. Sen, *Algorithms for stochastic mixed-integer programming models*, in Handbook in Operations Research and Management Science: Discrete Optimization, K. Aardal, G. Nemhauser, and R. Weismantel, eds., Elsevier, 2005, pp. 515–558.

[19] S. Sen and J. L. Higle, *The C^3 theorem and a D^2 algorithm for large scale stochastic mixed-integer programming: set convexification*, Mathematical Programming, 104 (2005), pp. 1–20.

[20] S. Sen and H. D. Sherali, *On the convergence of cutting plane algorithms for a class of nonconvex mathematical programs*, Mathematical Programming, 106 (2006), pp. 203–223.

[21] H. D. Sherali and X. Zhu, *On solving discrete two-stage stochastic programs having mixed-integer first- and second- stage variables*, Mathematical Programming, 108 (2006), pp. 597–616.

[22] A. Trapp, O. A. Prokopyev, and A. J. Schaefer, *On a level-set characterization of the value function of an integer program and its application to stochastic programming*, Operations Research, 61 (2013), pp. 498–511.

[23] R. Van Slyke and R. Wets, *L-shaped linear programs with applications to optimal control and stochastic programming*, SIAM Journal on Applied Mathematics, 17 (1969), pp. 638–663.