Nonconvex Generalized Benders Decomposition for Stochastic Separable Mixed-Integer Nonlinear Programs

Citation: Li, Xiang et al. "Nonconvex generalized benders decomposition for stochastic separable mixed-integer nonlinear programs." Journal of Optimization Theory and Applications 151 (December 2011): 425 © 2011 Publisher

As Published: http://dx.doi.org/10.1007/s10957-011-9888-1

Publisher: Springer Science and Business Media LLC

Persistent URL: https://hdl.handle.net/1721.1/122814

Version: Author's final manuscript: final author's manuscript post peer review, without publisher's formatting or copy editing

Noname manuscript No.
(will be inserted by the editor)

Nonconvex Generalized Benders Decomposition for Stochastic Separable Mixed-Integer Nonlinear Programs

Xiang Li · Asgeir Tomasgard · Paul I. Barton

Communicated by Panos M. Pardalos

Abstract This paper considers deterministic global optimization of scenario-based, two-stage stochastic mixed-integer nonlinear programs (MINLPs) in which the participating functions are nonconvex and separable in integer and continuous variables. A novel decomposition method based on generalized Benders decomposition, named nonconvex generalized Benders decomposition (NGBD), is developed to obtain ε-optimal solutions of the stochastic MINLPs of interest in finite time. The dramatic computational advantage of NGBD over state-of-the-art global optimizers is demonstrated through the computational study of several engineering problems, where a problem with almost 150,000 variables is solved by NGBD within 80 minutes of solver time.

This work was supported by Statoil and the research council of Norway (project nr176089/S60) as part of the paired Ph.D. research program in gas technologies between MIT and NTNU.

The final publication is available at Springer via http://dx.doi.org/10.1007/s10957-011-9888-1

Xiang Li
Process Systems Engineering Laboratory, Department of Chemical Engineering, Massachusetts Institute of Technology, Cambridge, MA, USA

Asgeir Tomasgard
Department of Industrial Economics and Technology Management, Norwegian University of Science and Technology, Trondheim, Norway

Paul I. Barton (Corresponding Author)
Process Systems Engineering Laboratory, Department of Chemical Engineering, Massachusetts Institute of Technology, Cambridge, MA, USA
Tel.: +1-617-253-6526
Fax: +1-617-258-5042
E-mail: pib@mit.edu

Keywords Stochastic programming · Mixed-integer nonlinear programming · Decomposition algorithm · Global optimization

Mathematics Subject Classification (2000) 90C26 · 90C15

1 Introduction

Mixed-integer nonlinear programs (MINLPs) provide a powerful framework for mathematical modeling of problems that involve discrete and continuous decisions and nonlinearities. Over the past several decades there has been a tremendous amount of work on the development and solution of MINLP models in various engineering areas [1] [2], from product and process design to process operation and control [3]. These problems have been traditionally solved with deterministic models, although the real systems are almost always uncertain. Recently, more and more attention has been paid to including uncertainties into the optimization models [4], especially when the uncertainties have a significant impact on the decision made. Stochastic programming with recourse [5] is a natural way to address uncertainties in various engineering problems, such as natural gas production network design [6], oil or gas field infrastructure planning [7], water network synthesis [8], optimal storage design [9], chemical process synthesis [10], capacity expansion [11], etc.

This paper is devoted to scenario-based, two-stage stochastic MINLPs with recourse, whose deterministic equivalent programs exhibit the following structure:

$$\begin{aligned} \min_{x_1,\ldots,x_s,y} \ & \sum_{h=1}^{s} w_h \left(c_h^{\mathrm{T}} y + f_h(x_h) \right) \\ \text{s.t.} \ & g_h(x_h) + B_h y \le 0, \quad \forall h \in \{1,...,s\}, \\ & x_h \in X_h, \quad \forall h \in \{1,...,s\}, \\ & y \in Y, \end{aligned} \tag{P}$$

where $X_h = \{x_h \in \Pi_h \subset \mathbb{R}^{n_x} : p_h(x_h) \le 0\}$, $Y = \{y \in \{0,1\}^{n_y} : Ay \le d\}$, Π_h is convex, functions $f_h : \Pi_h \to \mathbb{R}$, $g_h : \Pi_h \to \mathbb{R}^m$ and $p_h : \Pi_h \to \mathbb{R}^{m_p}$ are continuous, and it is assumed at least one function in Problem (P) is nonconvex. The uncertainties are characterized by s different uncertainty realizations, also called *scenarios* [5] [12], which are indexed by h.

The binary variables y represent first-stage decisions which are made before realization of the uncertainties, while the continuous variables x_h include second-stage decisions made after the outcome of scenario h and the corresponding dependent variables. $c_h^{\mathrm{T}} y$ represents the cost associated with the first-stage decisions for scenario h and $f_h(x_h)$ represents the cost associated with the second-stage decisions for scenario h. $w_h > 0$ represents the probability of the occurrence of scenario h. Notice that if there are any linear or nonlinear equality constraints in the problem, they can be written as paired inequalities. Obviously, the size of Problem (P) depends on the number of scenarios addressed, s. When s is large, Problem (P) is a large-scale MINLP even if the model with one scenario is small.

Due to their special structures, two-stage stochastic programs with recourse [5] have long been solved with duality-based decomposition methods. The advantage of the duality-based decomposition methods is that the sizes of the subproblems to be solved are independent of the number of scenarios addressed, and most of the subproblems can be solved in parallel. One class of such methods is Benders decomposition (BD) [13], also called the *L*-shaped method in the stochastic programming literature [14], and its extension to nonlinear problems, generalized Benders decomposition (GBD) [15]. In BD or GBD, the original problem is projected onto the space of first-stage variables and reformulated into a dual problem that contains an infinite number of constraints, which is then relaxed into a lower bounding problem with a finite subset of these constraints. After fixing the first-stage variables to the solution of the lower bounding problem, the original problem becomes an upper bounding problem, which can naturally be decomposed into smaller subproblems for each of the scenarios. The solutions of a sequence of upper bounding problems give a sequence of nondecreasing upper bounds on the optimal objective function value while the solutions of a sequence of lower bounding problems give a sequence of nonincreasing lower bounds. An optimum of the original problem is obtained when the upper and lower bounds converge. Such convergence relies on strong duality during the dual reformulation and it is not guaranteed for many nonconvex problems.

Another class of duality-based decomposition methods is Lagrangian decomposition [16] [17]. In Lagrangian decomposition, the first-stage variables are duplicated for each of the scenarios and the variables in different scenarios are linked with additional equality constraints. The dual of this problem, which is generated by dualizing the linking constraints into the objective function, is convex but nonsmooth. This nonsmooth problem is usually solved with a subgradient method (e.g., [18]), in which each subproblem can naturally be decomposed into smaller subproblems for different scenarios. Since Lagrangian decomposition is usually applied to nonconvex problems for which strong duality does not hold, this method is usually performed within a branch-and-bound framework to guarantee convergence to a global optimum. For example, Karuppiah and Grossmann [19] have developed a branch-and-cut method for solving a class of stochastic MINLP problems, where Lagrangian decomposition is used to generate cuts to strengthen the subproblems obtained through convex relaxation. Khajavirad and Michalek [20] have developed a similar method for a class of quasiseparable MINLP problems. However, convergence of the upper and lower bounds in both methods can only be guaranteed by branching in the full variable space, whose dimension depends on the number of scenarios addressed. Therefore, these methods may not be able to solve problems with large numbers of scenarios rigorously within reasonable time.

As a nonconvex MINLP, Problem (P) can be solved rigorously by general-purpose deterministic global optimization methods, such as branch-and-reduce [21], SMIN-αBB and GMIN-αBB [22], and nonconvex outer approximation [23] (which is an extension of traditional outer approximation methods [24] [25] to programs with nonconvex functions participating). However, these methods cannot fully exploit the decomposable structure of Problem (P), i.e., they cannot solve Problem (P) via solving subproblems whose sizes are all independent of the number of scenarios, so they are usually not practical for problems with large numbers of scenarios.

This paper presents an extension of GBD to handle nonconvexity in Problem (P) rigorously, which can obtain an ε-optimal solution for Problem (P) in finite time. The new GBD method is termed nonconvex generalized Benders decomposition (NGBD). The remaining part of the paper is organized as follows: Section 2 introduces the general idea of NGBD along with the reformulation of Problem (P) and the resulting subproblems. Section 3 proves the important properties of the subproblems and Sections 4 gives the NGBD algorithm with a finite convergence result. Section 5 presents the case study results which demonstrate the computational advantage of NGBD over state-of-the-art global optimizers for stochastic MINLP. The paper ends with concluding remarks in Section 6.

2 Reformulation and the Subproblems

GBD can be viewed as a result of applying the framework of concepts presented by Geoffrion for the design of large-scale mathematical programming techniques [26] [27]. The framework includes two groups of concepts: problem manipulations and solution strategies. Problem manipulations, including projection, dualization, inner linearization and outer linearization, are devices for restating a given problem in an alternative form more amenable to solution. The result is often what is referred to as a master problem. Solution strategies, including relaxation, restriction and piecewise linearization, reduce the master problem to a related sequence of simpler subproblems. GBD employs the concepts of projection, dualization, restriction and relaxation; for nonconvex problems, the dualization manipulation may affect the convergence property of the method because of duality gap. The NGBD proposed in this paper extends GBD to solve problem with nonconvex functions participating, by adding an additional convex lower bounding problem as a surrogate of the original problem. The traditional GBD iteration is applied to the convex lower bounding problem instead of the original problem, which not only yields a sequence of valid lower bounds for the original problem, but also gives hints to construct a sequence of primal subproblems that provide valid upper bounds for the original problem.

Before the lower bounding problem and the other subproblems in nonconvex GBD are detailed, the following assumptions are made:

Assumption 2.1. *Set Y is nonempty.*

Assumption 2.2. *Set X_h is nonempty and compact for any $h \in \{1, \ldots, s\}$.*

Remark 2.1 *Assumption 2.2 implies that the feasible set of Problem (P) is compact for y fixed to any element in Y. So Problem (P) either has finite optimal objective value or is infeasible due to the continuity of the functions therein.*

2.1 Convexification - Lower Bounding Problem

Definition 2.1 (Convex Relaxation) Given convex sets Π_h and $\Theta_h \subset \mathbb{R}^{n_q}$, continuous functions $u_{f,h} : \Pi_h \times \Theta_h \to \mathbb{R}$, $u_{g,h} : \Pi_h \times \Theta_h \to \mathbb{R}^m$, $u_{p,h} : \Pi_h \times \Theta_h \to \mathbb{R}^{m_p}$, $u_{q,h} : \Pi_h \times \Theta_h \to \mathbb{R}^{m_q}$ define convex relaxations of functions f_h, g_h, p_h, q_h, if:
(1) $u_{f,h}$, $u_{g,h}$, $u_{p,h}$ and $u_{q,h}$ are convex on $\Pi_h \times \Theta_h$;
(2) For any $\hat{x}_h \in \Pi_h$, there exists $\hat{q}_h \in \Theta_h$ so that $u_{f,h}(\hat{x}_h, \hat{q}_h) \leq f_h(\hat{x}_h)$, $u_{g,h}(\hat{x}_h, \hat{q}_h) \leq g_h(\hat{x}_h)$, $u_{p,h}(\hat{x}_h, \hat{q}_h) \leq p_h(\hat{x}_h)$ and $u_{q,h}(\hat{x}_h, \hat{q}_h) \leq 0$.

The convex relaxations involve additional variables q_h and constraints $u_{q,h}(x_h, q_h) \leq 0$ that may be required to construct differentiable relaxations. Several convex relaxation techniques are available to generate convex relaxations, e.g., McCormick relaxation [28] and outer linearization [21] for factorable nonconvex functions, which usually introduce additional variables and constraints for differentiable relaxations, and αBB for twice-differentiable nonconvex functions [29], which does not require additional variables and constraints. Readers can refer to [30] for more discussions on the convex relaxation techniques.

Since Problem (P) is separable in the continuous and the integer variables, the continuous and discrete feasible regions can be individually characterized [24]. So it suffices to replace the nonconvex functions in Problem (P) with their convex relaxations to yield a *lower bounding problem* in the following form:

$$\begin{aligned} \min_{\substack{x_1,\ldots,x_s,\\ q_1,\ldots,q_s,y}} \quad & \sum_{h=1}^{s} w_h \left(c_h^{\mathrm{T}} y + u_{f,h}(x_h, q_h) \right) \\ \text{s.t.} \quad & u_{g,h}(x_h, q_h) + B_h y \leq 0, \quad \forall h \in \{1, \ldots, s\}, \\ & (x_h, q_h) \in D_h, \quad \forall h \in \{1, \ldots, s\}, \\ & y \in Y, \end{aligned} \tag{LBP}$$

where $D_h = \{(x_h, q_h) \in \Pi_h \times \Theta_h : u_{p,h}(x_h, q_h) \leq 0, u_{q,h}(x_h, q_h) \leq 0\}$. Obviously, Problem (LBP) is convex. If the functions f_h, g_h, p_h in Problem (P) are all convex, Problem (LBP) is equivalent to Problem (P) and the NGBD method reduces to the traditional GBD method.

Assumption 2.3. *Set D_h is compact for any $h \in \{1, \ldots, s\}$.*

Remark 2.2 *Assumption 2.3 implies that the feasible set of Problem (LBP) is compact for y fixed to any element in Y. So Problem (LBP) either has finite optimal objective value or is infeasible due to the continuity of the functions therein.*

Assumption 2.4. *Problem (LBP) satisfies Slater's condition for y fixed to those elements in Y for which Problem (LBP) is feasible.*

Remark 2.3 *Assumption 2.4 implies that strong duality holds for Problem (LBP) for y fixed to those elements in Y for which Problem (LBP) is feasible. This validates the dualization manipulation of the problem to a master problem.*

Remark 2.4 *Linear equalities may appear in Problem (LBP) as paired linear inequalities. For this case, a refined Slater's condition [31] for Problem (LBP) (to guarantee strong duality) can be stated as: For Problem (LBP) for y fixed to any element in Y for which Problem (LBP) is feasible, there exist $(\hat{x}_h, \hat{q}_h) \in relint(\Pi_h \times \Theta_h)$ $(h = 1, ..., s)$ for which all the linear constraints of Problem (LBP) are satisfied and all the nonlinear inequalities of the problem are strictly satisfied. Here $relint(\cdot)$ denotes the relative interior of a set.*

2.2 Projection/Dualization - Master Problem

Problem (LBP) is potentially a large-scale convex MINLP because the number of its continuous variables depends on the number of scenarios addressed in the problem. According to the principle of projection explained in [15], Problem (LBP) can be projected from the space of both the continuous and integer variables to the space of the integer variables, and any subproblem with a fixed integer realization can be reformulated into its dual. Thus, Problem

(LBP) can be transformed into the following *master problem*:

$$
\begin{aligned}
\min_{\eta,y} \quad & \eta \\
\text{s.t.} \;\; & \eta \geq \sum_{h=1}^{s} \inf_{(x_h,q_h)\in D_h} \left[w_h u_{f,h}(x_h,q_h) + \lambda_h^{\mathrm{T}} u_{g,h}(x_h,q_h) \right] + \left(\sum_{h=1}^{s} \left(w_h c_h^{\mathrm{T}} + \lambda_h^{\mathrm{T}} B_h \right) \right) y, \\
& \quad \forall \lambda_1,...,\lambda_s \geq 0 \\
& 0 \geq \sum_{h=1}^{s} \inf_{(x_h,q_h)\in D_h} \left[\mu_h^{\mathrm{T}} u_{g,h}(x_h,q_h) \right] + \left(\sum_{h=1}^{s} \mu_h^{\mathrm{T}} B_h \right) y, \quad \forall (\mu_1,...,\mu_s) \in M1, \\
& y \in Y, \; \eta \in \mathbb{R},
\end{aligned}
\tag{MP1}
$$

where

$$
\lambda_h \in \mathbb{R}^m, \quad \mu_h \in \mathbb{R}^m, \quad h = 1,...,s,
$$

and

$$
M1 = \{(\mu_1,...,\mu_s) : \mu_1,...,\mu_s \geq 0, \sum_{h=1}^{s} \sum_{i=1}^{m} \mu_{h,i} = 1\}.
$$

For convenience of establishing valid subproblems later, Problem (MP1) is further reformulated into the following form (by replacing set $M1$ with set M):

$$
\begin{aligned}
\min_{\eta,y} \quad & \eta \\
\text{s.t.} \;\; & \eta \geq \sum_{h=1}^{s} \inf_{(x_h,q_h)\in D_h} \left[w_h u_{f,h}(x_h,q_h) + \lambda_h^{\mathrm{T}} u_{g,h}(x_h,q_h) \right] + \left(\sum_{h=1}^{s} \left(w_h c_h^{\mathrm{T}} + \lambda_h^{\mathrm{T}} B_h \right) \right) y, \\
& \quad \forall \lambda_1,...,\lambda_s \geq 0 \\
& 0 \geq \sum_{h=1}^{s} \inf_{(x_h,q_h)\in D_h} \left[\mu_h^{\mathrm{T}} u_{g,h}(x_h,q_h) \right] + \left(\sum_{h=1}^{s} \mu_h^{\mathrm{T}} B_h \right) y, \quad \forall (\mu_1,...,\mu_s) \in M, \\
& y \in Y, \; \eta \in \mathbb{R},
\end{aligned}
\tag{MP}
$$

where

$$
M = \{(\mu_1,...,\mu_s) : \mu_1,...,\mu_s \geq 0, \sum_{h=1}^{s} \sum_{i=1}^{m} \mu_{h,i} > 0\}.
$$

The equivalence of Problems (MP1) and (MP) will be proved in the next section.

2.3 Restriction - Primal Problem, Primal Bounding Problem and Feasibility Problem

The *primal problem* is obtained through restricting y in Problem (P) to an element $y^{(l)}$ in Y, where the superscript l enumerates the sequence of integer realizations visited by the primal problem (i.e., the integer realizations for which the primal problem was solved). This problem can be written as follows:

$$\begin{aligned} obj_{\mathrm{PP}}(y^{(l)}) = \min_{x_1,\ldots,x_s} & \sum_{h=1}^{s} w_h \left(c_h^{\mathrm{T}} y^{(l)} + f_h(x_h) \right) \\ \text{s.t. } & g_h(x_h) + B_h y^{(l)} \leq 0, \quad \forall h \in \{1,\ldots,s\}, \\ & x_h \in X_h, \quad \forall h \in \{1,\ldots,s\}, \end{aligned} \tag{PP^l}$$

where $obj_{\mathrm{PP}}(y^{(l)})$ denotes the optimal objective value of Problem (PP^l) (which depends on the integer realization $y^{(l)}$). Problem (PP^l) can naturally be decomposed into subproblems for each of the s scenarios as follows:

$$\begin{aligned} obj_{\mathrm{PP}_h}(y^{(l)}) = \min_{x_h} & \, w_h \left(c_h^{\mathrm{T}} y^{(l)} + f_h(x_h) \right) \\ \text{s.t. } & g_h(x_h) + B_h y^{(l)} \leq 0, \\ & x_h \in X_h, \end{aligned} \tag{PP$_h^l$}$$

where $obj_{\mathrm{PP}_h}(y^{(l)})$ denotes the optimal objective value of Problem (PP$_h^l$), $h = 1,\ldots,s$. Obviously, $obj_{\mathrm{PP}}(y^{(l)}) = \sum_{h=1}^{s} obj_{\mathrm{PP}_h}(y^{(l)})$.

Remark 2.5 *The nonlinear programming (NLP) problem (PP$_h^l$) can be solved to ε-optimality in finite time by state-of-the-art global optimization solvers, such as BARON [21] [32], provided suitable convex underestimators of the participating functions can be constructed.*

Similarly, the *primal bounding problem* is obtained through restricting y in Problem (LBP) to an element $y^{(k)}$ in Y, where the superscript k enumerates the sequence of integer realizations visited by the primal bounding problem. This problem can be written as follows:

$$
\begin{aligned}
obj_{\mathrm{PBP}}(y^{(k)}) = \min_{\substack{x_1,\dots,x_s,\\ q_1,\dots,q_s}} & \sum_{h=1}^{s} w_h \left(c_h^{\mathrm{T}} y^{(k)} + u_{f,h}(x_h, q_h) \right) \\
\text{s.t. } & u_{g,h}(x_h, q_h) + B_h y^{(k)} \le 0, \quad \forall h \in \{1,...,s\}, \\
& (x_h, q_h) \in D_h, \quad \forall h \in \{1,...,s\},
\end{aligned}
\tag{PBP^k}
$$

where $obj_{\mathrm{PBP}}(y^{(k)})$ denotes the optimal objective value of Problem (PBP^k). The primal bounding problem can naturally be decomposed into subproblems for each of the s scenarios as follows:

$$
\begin{aligned}
obj_{\mathrm{PBP}_h}(y^{(k)}) = \min_{x_h, q_h} & \, w_h (c_h^{\mathrm{T}} y^{(k)} + u_{f,h}(x_h, q_h)) \\
\text{s.t. } & u_{g,h}(x_h, q_h) + B_h y^{(k)} \le 0, \\
& (x_h, q_h) \in D_h,
\end{aligned}
\tag{PBP$_h^k$}
$$

where $obj_{\mathrm{PBP}_h}(y^{(k)})$ denotes the optimal objective value of Problem (PBP$_h^k$), $h = 1,...,s$. Obviously, $obj_{\mathrm{PBP}}(y^{(k)}) = \sum_{h=1}^{s} obj_{\mathrm{PBP}_h}(y^{(k)})$.

If Problem (PBP$_h^k$) is infeasible for a scenario, Problem (PBP^k) is infeasible. Then the following *feasibility problem* is solved:

$$
\begin{aligned}
obj_{\mathrm{FP}}(y^{(k)}) = \min_{\substack{x_1,\dots,x_s,q_1,\dots,q_s,\\ z_1,\dots,z_s}} & \sum_{h=1}^{s} w_h ||z_h|| \\
\text{s.t. } & u_{g,h}(x_h, q_h) + B_h y^{(k)} \le z_h, \quad \forall h \in \{1,...,s\}, \\
& (x_h, q_h) \in D_h, \quad z_h \in Z_h, \quad \forall h \in \{1,...,s\},
\end{aligned}
\tag{FP^k}
$$

where $obj_{\mathrm{FP}}(y^{(k)})$ denotes the optimal objective value of Problem (FP^k), $||z_h||$ denotes an arbitrary norm of the slack variable vector z_h for $h = 1,...,s$, set $Z_h \subset \{z \in \mathbb{R}^m : z \ge 0\}$ and it has three additional properties (for $h = 1,...,s$):

(1) Z_h is a convex set;

(2) Z_h is a pointed cone, i.e., $0 \in Z_h$, and $\forall \alpha > 0, z \in Z_h$ implies $\alpha z \in Z_h$;

(3) There exists $\hat{z} \in Z_h$ such that $\hat{z} > 0$ (therefore the cone Z_h is unbounded from above in each dimension).

Each element of z measures the violation of a constraint, so the norm of z is minimized for minimum violation of the constraints. Since any norm function is convex, Problem (FP^k) is convex. Again, Problem (FP^k) can naturally be decomposed into convex subproblems for each of the s scenarios as follows:

$$
\begin{aligned}
obj_{\mathrm{FP}_h}(y^{(k)}) = \min_{x_h, q_h, z_h} \; & w_h ||z_h|| \\
\text{s.t. } \; & u_{g,h}(x_h, q_h) + B_h y^{(k)} \leq z_h, \qquad (\mathrm{FP}_h^k) \\
& (x_h, q_h) \in D_h, \quad z_h \in Z_h,
\end{aligned}
$$

where $obj_{\mathrm{FP}_h}(y^{(k)})$ denotes the optimal objective value of Problem (FP$_h^k$), $h = 1, ..., s$, and $\sum_{h=1}^{s} obj_{\mathrm{FP}_h}(y^{(k)}) = obj_{\mathrm{FP}}(y^{(k)})$.

Remark 2.6 *If the convex subproblems (PBP$_h^k$) and (FP$_h^k$) are smooth, they can be solved by gradient-based optimization solvers such as CONOPT [33], SNOPT [34], CPLEX [35] (only for linear programs, convex quadratic programs and quadratically constrained programs). Otherwise, they may be solved by nonsmooth optimization methods such as bundle methods [36].*

Remark 2.7 *When the number of scenarios addressed is large, w_h may be so small that the optimal objective values of Problems (PP$_h^l$), (PBP$_h^k$) and (FP$_h^k$) are smaller than the tolerance set for the optimization (for some scenarios). To avoid such ill-conditioning in practice, these problems can be solved without multiplying the cost function by w_h.*

2.4 Relaxation - Relaxed Master Problem

Although the number of decision variables in the master problem is independent of the number of scenarios in the original problem, the master problem (MP) is still difficult to solve directly because of the infinite number of constraints. For easier solution, Problem (MP) is relaxed at the kth GBD iteration into the following *relaxed master problem* with a finite number of constraints:

$$
\begin{aligned}
\min_{\eta,y}\quad & \eta \\
\text{s.t.}\quad & \eta \geq \sum_{h=1}^{s} \inf_{(x_h,q_h)\in D_h} \left[w_h u_{f,h}(x_h,q_h) + \left(\lambda_h^{(j)}\right)^{\mathrm{T}} u_{g,h}(x_h,q_h) \right] \\
& \quad + \left(\sum_{h=1}^{s} \left(w_h c_h^{\mathrm{T}} + \left(\lambda_h^{(j)}\right)^{\mathrm{T}} B_h \right) \right) y, \quad \forall j \in T^k, \\
& 0 \geq \sum_{h=1}^{s} \inf_{(x_h,q_h)\in D_h} \left[\left(\mu_h^{(i)}\right)^{\mathrm{T}} u_{g,h}(x_h,q_h) \right] + \left(\sum_{h=1}^{s} \left(\mu_h^{(i)}\right)^{\mathrm{T}} B_h \right) y, \quad \forall i \in S^k, \qquad (\mathrm{RMP1}^k) \\
& \sum_{r\in\{r:y_r^{(t)}=1,\, r=1,\dots,n_y\}} y_r - \sum_{r\in\{r:y_r^{(t)}=0,\, r=1,\dots,n_y\}} y_r \leq |\{r : y_r^{(t)} = 1\}| - 1, \\
& \quad \forall t \in T^k \cup S^k, \\
& y \in Y,\ \eta \in \mathbb{R},
\end{aligned}
$$

where the index sets

$$T^k = \{j \in \{1,\dots,k\} : \text{Problem (PBP) is feasible for } y = y^{(j)}\},$$

$$S^k = \{i \in \{1,\dots,k\} : \text{Problem (PBP) is infeasible for } y = y^{(i)}\}.$$

$\lambda_h^{(j)}$ are the Lagrange multipliers for Problem (PBP$_h^j$), which form an optimality cut for iteration j ($\forall j \in T^k$). $\mu_h^{(i)}$ are the Lagrange multipliers for Problem (FP$_h^i$), which form a feasibility cut for iteration i ($\forall i \in S^k$). The additional constraints in Problem (RMP1^k), which do not appear in the master problem (MP) stated before, represent a set of canonical integer cuts that prevent the previously examined integer realizations from becoming a solution [37].

Definition 2.2 λ^* is a Lagrange multiplier for the optimization problem

$$
\begin{aligned}
\min_{x}\ & f(x) \\
\text{s.t.}\quad & g(x) \leq 0, \\
& x \in X,
\end{aligned}
$$

if $\lambda^* \geq 0$ and $f(x^*) = \inf_{x \in X} \left[f(x) + (\lambda^*)^{\mathrm{T}} g(x) \right]$, where x^* denotes an optimal solution of the problem.

Remark 2.8 *Definition 2.2 for Lagrange multipliers follows from [38] in the context of duality theory (where they are called geometric multipliers instead). This definition is consistent with the one used by Geoffrion (where they are called optimal multipliers) for the GBD method [15] and duality theory [39]. Note that the Lagrange multipliers defined here are in general different from the multipliers that satisfy the Karush-Kuhn-Tucker (KKT) conditions, which are usually called KKT multipliers. However, for the convex program (PBP$_h^k$) or (FP$_h^k$) discussed in this paper, KKT multipliers are also Lagrange multipliers, as implied by the theorem on page 211 of [40]. State-of-the-art optimization solvers, such as CONOPT, SNOPT, CPLEX, offer such multiplier values at the solution, so there is no need to develop an additional algorithm to obtain the Lagrange multipliers for Problem (PBP$_h^k$) or (FP$_h^k$) in NGBD.*

When $T^k = \emptyset$, Problem (RMP1^k) is unbounded; in this case, the following feasibility relaxed master problem is solved instead:

$$\begin{aligned}
\min_{y} \quad & \sum_{i=1}^{n_y} y_i \\
\text{s.t.} \quad & 0 \geq \sum_{h=1}^{s} \inf_{(x_h, q_h) \in D_h} \left[\left(\mu_h^{(i)}\right)^{\mathrm{T}} u_{g,h}(x_h, q_h) \right] + \left(\sum_{h=1}^{s} \left(\mu_h^{(i)}\right)^{\mathrm{T}} B_h \right) y, \quad \forall i \in S^k, \\
& \sum_{r \in \{r : y_r^{(t)} = 1,\, r = 1, \ldots, n_y\}} y_r - \sum_{r \in \{r : y_r^{(t)} = 0,\, r = 1, \ldots, n_y\}} y_r \leq |\{r : y_r^{(t)} = 1\}| - 1, \quad \forall t \in S^k, \\
& y \in Y.
\end{aligned} \tag{FRMP1^k}$$

As will be demonstrated in the next section, the inner optimization problems in Problems (RMP1^k) and (FRMP1^k) can be removed according to the solution of the previously solved primal bounding problems and feasibility problems. Then Problem (RMP1^k) is equivalent to the following single-level mixed-integer linear program (MILP):

$$
\begin{aligned}
\min_{\eta,y} \quad & \eta \\
\text{s.t.} \quad & \eta \geq obj_{\mathrm{PBP}}(y^{(j)}) + \left(\sum_{h=1}^{s} \left(w_h c_h^{\mathrm{T}} + \left(\lambda_h^{(j)}\right)^{\mathrm{T}} B_h \right) \right) \left(y - y^{(j)} \right), \quad \forall j \in T^k, \\
& 0 \geq obj_{\mathrm{FP}}(y^{(i)}) + \left(\sum_{h=1}^{s} \left(\mu_h^{(i)}\right)^{\mathrm{T}} B_h \right) \left(y - y^{(i)} \right), \quad \forall i \in S^k, \\
& \sum_{r \in \{r : y_r^{(t)} = 1,\, r=1,\ldots,n_y\}} y_r - \sum_{r \in \{r : y_r^{(t)} = 0,\, r=1,\ldots,n_y\}} y_r \leq |\{r : y_r^{(t)} = 1\}| - 1, \\
& \qquad \forall t \in T^k \cup S^k, \\
& y \in Y, \; \eta \in \mathbb{R},
\end{aligned}
\tag{RMP^k}
$$

and Problem (FRMP1^k) is:

$$
\begin{aligned}
\min_{y} \quad & \sum_{i=1}^{n_y} y_i \\
\text{s.t.} \quad & 0 \geq obj_{\mathrm{FP}}(y^{(i)}) + \left(\sum_{h=1}^{s} \left(\mu_h^{(i)}\right)^{\mathrm{T}} B_h \right) \left(y - y^{(i)} \right), \quad \forall i \in S^k, \\
& \sum_{r \in \{r : y_r^{(t)} = 1,\, r=1,\ldots,n_y\}} y_r - \sum_{r \in \{r : y_r^{(t)} = 0,\, r=1,\ldots,n_y\}} y_r \leq |\{r : y_r^{(t)} = 1\}| - 1, \quad \forall t \in S^k \\
& y \in Y.
\end{aligned}
\tag{FRMP^k}
$$

Notice that the size of Problem (RMP^k) or Problem (FRMP^k) is independent of the number of the scenarios in the original problem.

2.5 Summary

This section details the reformulation of the original stochastic MINLP into a collection of subproblems through convexification, projection/dualization, restriction and relaxation. Figure 1 summarizes these subproblems and the way to obtain them. The subproblems in boxes with dashed lines are the intermediate subproblems for the reformulation which are not solved directly in NGBD. The subproblems in boxes with solid lines are solved directly in NGBD (by calling appropriate optimization solvers), including relaxed master problem

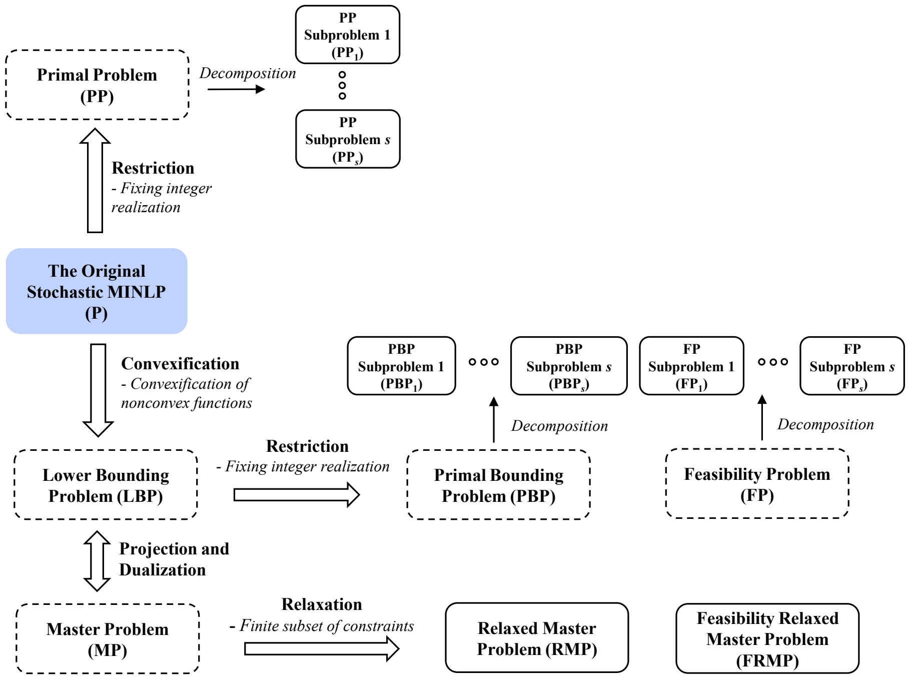
Primal Problem (PP)
Decomposition
PP Subproblem 1 (PP$_1$)
PP Subproblem s (PP$_s$)
Restriction
- Fixing integer realization
The Original Stochastic MINLP (P)
Convexification
- Convexification of nonconvex functions
PBP Subproblem 1 (PBP$_1$)
PBP Subproblem s (PBP$_s$)
FP Subproblem 1 (FP$_1$)
FP Subproblem s (FP$_s$)
Decomposition
Decomposition
Lower Bounding Problem (LBP)
Restriction
- Fixing integer realization
Primal Bounding Problem (PBP)
Feasibility Problem (FP)
Projection and Dualization
Master Problem (MP)
Relaxation
- Finite subset of constraints
Relaxed Master Problem (RMP)
Feasibility Relaxed Master Problem (FRMP)

Figure 1 Overview of the reformulation and the subproblems in nonconvex GBD.

and feasibility relaxed master problem, decomposed primal bounding subproblems and feasibility subproblems, and decomposed primal subproblems. Notice that the sizes of all the subproblems to be solved are independent of the number of the scenarios addressed by the original problem. Next, the important properties of the subproblems derived in this section will be outlined and discussed via a set of propositions in Section 3. Based on these results, the NGBD algorithm is developed with a convergence proof in Section 4.

3 Properties of the Subproblems

Proposition 3.1 *The optimal objective value of Problem (LBP) represents a lower bound on the optimal objective value of Problem (P).*

Proof. According to Definition 2.1 for the convex relaxations, for any $(\hat{x}_1,..,\hat{x}_s,\hat{y})$ that is feasible to Problem (P), there exists $\hat{q}_h \in \Theta_h$ which gives $u_{f,h}(\hat{x}_h,\hat{q}_h) \le f_h(\hat{x}_h) \le 0$, $u_{g,h}(\hat{x}_h,\hat{q}_h) \le g_h(\hat{x}_h) \le 0$, $u_{p,h}(\hat{x}_h,\hat{q}_h) \le p_h(\hat{x}_h) \le 0$ and $u_{q,h}(\hat{x}_h,\hat{q}_h) \le 0$ $(h = 1,...,s)$. So

$(\hat{x}_1,..,\hat{x}_s,\hat{q}_1,..,\hat{q}_s,\hat{y})$ is feasible to Problem (LBP) and it renders an optimal objective value $\sum_{h=1}^{s} w_h \left(u_{f,h}(\hat{x}_h,\hat{q}_h)+c_h^{\mathrm{T}}\hat{y}\right) \leq \sum_{h=1}^{s} w_h \left(f_h(\hat{x}_h)+c_h^{\mathrm{T}}\hat{y}\right)$ (noticing $w_h > 0$). Therefore the lower bounding property follows. □

Proposition 3.2 *Problems (LBP) and (MP1) are equivalent in the sense that:*

(1) Problem (LBP) is feasible iff Problem (MP1) is feasible;

(2) The optimal objective values of Problems (LBP) and (MP1) are the same;

(3) The optimal objective value of Problem (LBP) is attained with an integer realization iff the optimal objective value of Problem (MP1) is attained with the same integer realization.

Proof. Given Assumption 2.4, the results follow immediately from Theorems 2.1, 2.2 and 2.3 in [15]. □

Proposition 3.3 *Problems (MP1) and (MP) are equivalent in the sense that:*

(1) Problem (MP1) is feasible iff Problem (MP) is feasible;

(2) The optimal objective values of Problems (MP1) and (MP) are the same;

(3) The optimal objective value of Problem (MP1) is attained with an integer realization iff the optimal objective value of Problem (MP) is attained with the same integer realization.

Proof. The results can be proved by showing that Problems (MP1) and (MP) have the same feasible set. Denote the feasible regions of Problems (MP1) and (MP) by F_{MP1} and F_{MP}, respectively. $F_{MP1} = F_{MP}$ can be proved by showing $F_{MP} \subset F_{MP1}$ and $F_{MP1} \subset F_{MP}$.

First, for any $(\hat{y},\hat{\eta}) \in F_{MP}$,

$$\sum_{h=1}^{s} \inf_{(x_h,q_h)\in D_h} \left[\mu_h^{\mathrm{T}} u_{g,h}(x_h,q_h)\right] + \left(\sum_{h=1}^{s} \mu_h^{\mathrm{T}} B_h\right) y \leq 0, \quad \forall(\mu_1,...,\mu_s) \in M,$$

so

$$\sum_{h=1}^{s} \inf_{(x_h,q_h)\in D_h} \left[\mu_h^{\mathrm{T}} u_{g,h}(x_h,q_h)\right] + \left(\sum_{h=1}^{s} \mu_h^{\mathrm{T}} B_h\right) y \leq 0, \quad \forall(\mu_1,...,\mu_s) \in M1,$$

because $M1 \subset M$. Therefore, $F_{MP} \subset F_{MP1}$.

Second, for any $(\hat{y},\hat{\eta}) \in F_{MP1}$,

$$\sum_{h=1}^{s} \inf_{(x_h,q_h)\in D_h} \left[\mu_h^{\mathrm{T}} u_{g,h}(x_h,q_h)\right] + \left(\sum_{h=1}^{s} \mu_h^{\mathrm{T}} B_h\right) y \leq 0, \quad \forall(\mu_1,...,\mu_s) \in M1, \tag{1}$$

For such $(\hat{y},\hat{\eta})$, for any $(\hat{\mu}_1,...,\hat{\mu}_s) \in M$,

$$\sum_{h=1}^{s}\sum_{i=1}^{m}\hat{\mu}_{h,i} > 0, \tag{2}$$

so new multipliers $(\tilde{\mu}_1,...,\tilde{\mu}_s)$ can be defined as

$$\tilde{\mu}_{h,i} = \hat{\mu}_{h,i}/\left(\sum_{h=1}^{s}\sum_{i=1}^{m}\hat{\mu}_{h,i}\right), \quad \forall h \in \{1,...,s\},\ \forall i \in \{1,...,m\}, \tag{3}$$

then

$$(\tilde{\mu}_1,...,\tilde{\mu}_s) \in M1. \tag{4}$$

From (1) and (4),

$$\sum_{h=1}^{s}\inf_{(x_h,q_h)\in D_h}\left[\tilde{\mu}_h^{\mathrm{T}}u_{g,h}(x_h,q_h)\right] + \left(\sum_{h=1}^{s}\tilde{\mu}_h^{\mathrm{T}}B_h\right)y \le 0. \tag{5}$$

Considering (2), (3) and (5)

$$\begin{aligned}
&\sum_{h=1}^{s}\inf_{(x_h,q_h)\in D_h}\left[\hat{\mu}_h^{\mathrm{T}}u_{g,h}(x_h,q_h)\right] + \left(\sum_{h=1}^{s}\hat{\mu}_h^{\mathrm{T}}B_h\right)y \\
=&\left(\sum_{h=1}^{s}\sum_{i=1}^{m}\hat{\mu}_{h,i}\right)\left(\sum_{h=1}^{s}\inf_{(x_h,q_h)\in D_h}\left[\tilde{\mu}_h^{\mathrm{T}}u_{g,h}(x_h,q_h)\right] + \left(\sum_{h=1}^{s}\tilde{\mu}_h^{\mathrm{T}}B_h\right)y\right) \\
\le& 0
\end{aligned}$$

Therefore,

$$\sum_{h=1}^{s}\inf_{(x_h,q_h)\in D_h}\left[\mu_h^{\mathrm{T}}u_{g,h}(x_h,q_h)\right] + \left(\sum_{h=1}^{s}\mu_h^{\mathrm{T}}B_h\right)y \le 0, \quad \forall(\mu_1,...,\mu_s) \in M$$

as well, so $(\hat{y},\hat{\eta}) \in F_{MP}$ too. Thus, $F_{MP1} \subset F_{MP}$. □

Proposition 3.4 *For y fixed to any element in Y, if Problem (PP^l) is feasible, its optimal objective value is no less than the optimal objective value of Problem (P).*

Proof. This result trivially holds due to the construction of Problem (PP^l) and the principle of restriction. □

Proposition 3.5 *If the primal problem (PP^k) is feasible, the corresponding primal bounding problem (PBP^k) is feasible as well. In this case, the optimal objective value of Problem (PP^k) is no less than that of Problem (PBP^k). The same relationship holds for the decomposed primal subproblem (PP$_h^k$) and the decomposed primal bounding subproblem (PBP$_h^k$) for each scenario.*

Proof. This can be proved according to the construction of these problems and Definition 2.1 in the same way to prove Proposition 3.1. □

Remark 3.1 *Proposition 3.5 implies that, if the optimal objective value of Problem (PBP$_h^k$) is worse than that of Problem (P), there is no need to solve Problem (PP^k) because $y = y^{(k)}$ cannot lead to an optimum of Problem (P). This property will be exploited in the NGBD algorithm to reduce the number of the primal problems to be solved, since obtaining a global optimum for the primal problem is computationally expensive.*

Proposition 3.6 *Denote the known upper bound on the optimal objective value of Problem (P) by UBD. If Problem (PP^l) is feasible and $obj_{PP}(y^{(l)}) \leq UBD$, then*

$$obj_{PP_h}(y^{(l)}) \leq UBD_h = UBD - \sum_{i=1}^{h-1} obj_{PP_i}(y^{(l)}) - \sum_{j=h+1}^{s} obj_{PBP_j}(y^{(l)})$$

for any $h \in \{1,...,s\}$.

Proof. If Problem (PP^l) is feasible, Problem (PP$_h^l$) is feasible for any $h \in \{1,...,s\}$. Since

$$obj_{\mathrm{PP}}(y^{(l)}) = \sum_{h=1}^{s} obj_{\mathrm{PP}_h}(y^{(l)}),$$

$$obj_{\mathrm{PP}_h}(y^{(l)}) = obj_{\mathrm{PP}}(y^{(l)}) - \sum_{i=1}^{h-1} obj_{\mathrm{PP}_i}(y^{(l)}) - \sum_{j=h+1}^{s} obj_{\mathrm{PP}_j}(y^{(l)}),\ \forall h \in \{1,...,s\}.$$

According to Proposition 3.5,

$$obj_{\mathrm{PBP}_h}(y^{(l)}) \leq obj_{\mathrm{PP}_h}(y^{(l)}),\ \forall h \in \{1,...,s\},$$

so

$$obj_{\mathrm{PP}_h}(y^{(l)}) \leq UBD - \sum_{i=1}^{h-1} obj_{\mathrm{PP}_i}(y^{(l)}) - \sum_{j=h+1}^{s} obj_{\mathrm{PBP}_j}(y^{(l)}),\ \forall h \in \{1,...,s\}.$$

□

Remark 3.2 *Since $UBD_h = UBD - \sum_{i=1}^{h-1} obj_{PP_i}(y^{(l)}) - \sum_{j=h+1}^{s} obj_{PBP_j}(y^{(l)})$ can be calculated before solving Problem (PP$_h^l$), this value can be used to accelerate the global solution of Problem (PP$_h^l$) without excluding a solution that may lead to a global optimum of Problem (P). For example, this value can be used in branch-and-bound type solver, such as BARON, to fathom the nodes that do not contain solutions better than UBD_h.*

Proposition 3.7 *For any $h \in \{1,...,s\}$, Problem (FP$_h^k$) satisfies Slater's condition and it always has a minimum, say (x_h^*, q_h^*, z_h^*). $||z_h^*|| > 0$ for those scenarios in which Problem (PBP$_h^k$) is infeasible.*

Proof. According to Assumption 2.4, set D_h has at least one Slater point, say $(\hat{x}_h, \hat{q}_h)$. Due to the continuity of function $u_{g,h}$, $u_{g,h}(\hat{x}_h) + B_h y^{(k)}$ is finite, so there exists $\hat{z}_h \in Z_h$ such that $u_{g,h}(\hat{x}_h) + B_h y^{(k)} < \hat{z}_h$. Then $(\hat{x}_h, \hat{q_h}, \hat{z}_h)$ is a Slater point of Problem (FP$_h^k$), and strong duality holds for the problem. In addition, Problem (FP$_h^k$) has a closed feasible set and $||z_h||$ is continuous and coercive on Z_h, so Problem (FP$_h^k$) has a minimum according to Weierstrass' Theorem [38]. If Problem (PBP$_h^k$) is infeasible for scenario h, $z_h^* \neq 0$, so $||z_h^*|| > 0$. □

Proposition 3.8 *If μ_h^* are Lagrange multipliers for Problem (FP$_h^k$) ($h = 1,...,s$), then:*

(1) $\mu_h^* \geq 0$;

(2) $\inf_{(x_h,q_h)\in D_h} \left[(\mu_h^*)^{\mathrm{T}} \left(u_{g,h}(x_h,q_h) + B_h y^{(k)} \right) \right] > 0$ and $\sum_{i=1}^{m} \mu_{h,i}^* > 0$ for those scenarios in which Problem (PBP$_h^k$) is infeasible;

(3) $\mu_h^* = 0$ are valid Lagrange multiplier values for Problem (FP$_h^k$) for those scenarios in which Problem (PBP$_h^k$) is feasible;

(4) $(\mu_1^*, ..., \mu_s^*)$ are Lagrange multipliers for Problem (FP^k) and $\sum_{h=1}^{s} \sum_{i=1}^{m} \mu_{h,i}^* > 0$.

Proof. First, as Lagrange multipliers,

$$\mu_h^* \geq 0, \quad \forall h \in \{1,...,s\}. \tag{6}$$

Second, let (x_h^*, q_h^*, z_h^*) be a minimum of Problem (FP_h^k), then due to strong duality,

$$\begin{aligned} w_h||z_h^*|| &= \inf_{(x_h,q_h,z_h)\in D_h\times Z_h} \left[w_h||z_h|| + (\mu_h^*)^{\mathrm{T}}\left(u_{g,h}(x_h,q_h) + B_h y^{(k)} - z_h\right)\right] \\ &= \inf_{z_h\in Z_h}\left[w_h||z_h|| - (\mu_h^*)^{\mathrm{T}} z_h\right] + \inf_{(x_h,q_h)\in D_h}\left[(\mu_h^*)^{\mathrm{T}}\left(u_{g,h}(x_h,q_h) + B_h y^{(k)}\right)\right]. \end{aligned} \tag{7}$$

Suppose that

$$\inf_{z_h\in Z_h}\left[w_h||z_h|| - (\mu_h^*)^{\mathrm{T}} z_h\right] < 0, \tag{8}$$

then $\exists\varepsilon > 0$ such that

$$\inf_{z_h\in Z_h}\left[w_h||z_h|| - (\mu_h^*)^{\mathrm{T}} z_h\right] < -\varepsilon. \tag{9}$$

Hence, $\forall\alpha > 0$,

$$\alpha \inf_{z_h\in Z_h}\left[w_h||z_h|| - (\mu_h^*)^{\mathrm{T}} z_h\right] < -\alpha\varepsilon, \tag{10}$$

which is

$$\inf_{z_h\in Z_h}\left[w_h||\alpha z_h|| - (\mu_h^*)^{\mathrm{T}} (\alpha z_h)\right] < -\alpha\varepsilon. \tag{11}$$

Since $\forall z_h \in Z_h$, $\alpha z_h \in Z_h$ as well,

$$\inf_{z_h\in Z_h}\left[w_h||z_h|| - (\mu_h^*)^{\mathrm{T}} z_h\right] = \inf_{z_h\in Z_h}\left[w_h||\alpha z_h|| - (\mu_h^*)^{\mathrm{T}} (\alpha z_h)\right] < -\alpha\varepsilon \tag{12}$$

and therefore

$$\inf_{z_h\in Z_h}\left[w_h||z_h|| - (\mu_h^*)^{\mathrm{T}} z_h\right] = -\infty. \tag{13}$$

Equations (7) and (13) imply $w_h||z_h^*|| = -\infty$, so $||z_h^*|| = -\infty$ (since $w_h > 0$ as stated at the beginning of the paper), which contradicts the definition of a norm. Therefore, (8) is not true and

$$\inf_{z_h\in Z_h}\left[w_h||z_h|| - (\mu_h^*)^{\mathrm{T}} z_h\right] \geq 0. \tag{14}$$

When $z_h = 0$ $(\in Z_h)$, $w_h||z_h|| - (\mu_h^*)^{\mathrm{T}} z_h = 0$, so

$$\inf_{z_h\in Z_h}\left[w_h||z_h|| - (\mu_h^*)^{\mathrm{T}} z_h\right] \leq 0. \tag{15}$$

Inequalities (14) and (15) imply

$$\inf_{z_h \in Z_h} \left[w_h ||z_h|| - (\mu_h^*)^{\mathrm{T}} z_h \right] = 0. \tag{16}$$

Then, (7) and (16) imply

$$\inf_{(x_h,q_h)\in D_h} \left[(\mu_h^*)^{\mathrm{T}} \left(u_{g,h}(x_h,q_h) + B_h y^{(k)} \right) \right] = w_h ||z_h^*||. \tag{17}$$

If Problem (PBP$_h^k$) is infeasible, $||z_h^*|| > 0$, then Equation (17) implies

$$\inf_{(x_h,q_h)\in D_h} \left[(\mu_h^*)^{\mathrm{T}} \left(u_{g,h}(x_h,q_h) + B_h y^{(k)} \right) \right] > 0, \tag{18}$$

which further implies

$$\mu_h^* \neq 0. \tag{19}$$

Inequalities (6) and (19) imply

$$\sum_{i=1}^{m} \mu_{h,i}^* > 0. \tag{20}$$

Third, if Problem (PBP$_h^k$) is feasible for a scenario, the corresponding Problem (FP$_h^k$) has an optimal objective value of 0. According to Definition 2.2, $\mu_h^* = 0$ are valid Lagrange multiplier values for Problem (FP$_h^k$) for this scenario.

Finally, if a point (x_h^*, q_h^*, z_h^*) is a minimum of Problem (FP$_h^k$) for any $h \in \{1,...,s\}$, then the point $(x_1^*,...,x_s^*, q_1^*,...,q_s^*, z_1^*,...,z_s^*)$ is a minimum of problem (FP^k). According to (17),

$$\begin{aligned} \sum_{h=1}^{s} w_h ||z_h^*|| &= \sum_{h=1}^{s} \inf_{(x_h,q_h)\in D_h} \left[(\mu_h^*)^{\mathrm{T}} \left(u_{g,h}(x_h,q_h) + B_h y^{(k)} \right) \right] \\ &= \inf_{\substack{(x_h,q_h)\in D_h,\\ \forall h \in \{1,...,s\}}} \sum_{h=1}^{s} \left[(\mu_h^*)^{\mathrm{T}} \left(u_{g,h}(x_h,q_h) + B_h y^{(k)} \right) \right]. \end{aligned} \tag{21}$$

Therefore, $(\mu_1^*,...,\mu_s^*)$ are Lagrange multipliers for Problem (FP^k). In addition, solving Problem (FP^k) implies that the related Problem (PBP^k) is infeasible, i.e., the related Problem (PBP$_h^k$) is infeasible for at least one scenario and for this scenario $\sum_{i=1}^{m} \mu_{h,i}^* > 0$. Considering the nonnegativity of the Lagrange multipliers, $\sum_{h=1}^{s} \sum_{i=1}^{m} \mu_{h,i}^* > 0$ holds. □

Remark 3.3 *If Problem (PBP$_h^k$) is feasible for a scenario, the optimal objective value and one group of Lagrange multipliers of the corresponding Problem (FP$_h^k$) are known (i.e., all of them are zero), as indicated by (3) of Proposition 3.8. Therefore, there is no need to solve Problem (FP$_h^k$) for this scenario in NGBD.*

Proposition 3.9 *Problem (RMP1^k) is a relaxation of the master problem (MP) when (MP) is augmented with the relevant canonical integer cuts excluding the previously examined integer realizations.*

Proof. As Lagrange multipliers, $\lambda_1^{(j)}, ..., \lambda_s^{(j)} \geq 0$ $(\forall j \in T^k)$. According to Proposition 3.8, $(\mu_1^{(i)}, ..., \mu_s^{(i)}) \in M$ $(\forall i \in S^k)$. Therefore, Problem (RMP1^k) is a relaxation of the master problem (MP) excluding all the previously examined integer variables (i.e., augmented with the integer cuts). □

Proposition 3.10 *Problems (RMP1^k) and (RMP^k) are equivalent.*

Proof. This follows from the separability of the functions in the continuous and the integer variables. Detailed proof can be found in [15]. □

Corollary 3.1 *The relaxed master problems (RMP) and the feasibility relaxed master problem (FRMP) never generate the same integer solution twice.*

Corollary 3.2 *The optimal objective value of Problem (RMP^k) is a valid lower bound for the lower bounding problem (LBP) (or the master problem (MP)) augmented with the relevant canonical integer cuts and the original problem (P) augmented with the relevant canonical integer cuts.*

4 NGBD Algorithm

4.1 Algorithm

Initialize:

1. Iteration counters $k = 0$, $l = 1$, and the index sets $T^0 = \emptyset$, $S^0 = \emptyset$, $U^0 = \emptyset$.

```
2. Upper bound on the problem $UBD = +\infty$, upper bound on the lower bounding problem $UBDPB = +\infty$, lower bound on the lower bounding problem $LBD = -\infty$.
3. Set tolerances $\varepsilon_h$ and $\varepsilon$ such that $\sum_{h=1}^{s} \varepsilon_h \leq \varepsilon$.
4. Integer realization $y^{(1)}$ is given.
repeat
  if $k = 0$ or (Problem (RMP$^k$) is feasible and $LBD < UBDPB$ and $LBD < UBD - \varepsilon$)
  then
    repeat
      Set $k = k + 1$
      1. Solve the decomposed primal bounding subproblem (PBP$_h^k$) for each scenario $h = 1, ..., s$ sequentially. If Problem (PBP$_h^k$) is feasible and has duality multipliers $\lambda_h^{(k)}$ for all the scenarios, add an optimality cut to the relaxed master problem (RMP$^k$) according to the multipliers $\lambda_1^{(k)}, ..., \lambda_s^{(k)}$, set $T^k = T^{k-1} \cup \{k\}$. If $obj_{\text{PBP}}(y^{(k)}) = \sum_{h=1}^{s} obj_{\text{PBP}_h}(y^{(k)}) < UBDPB$, update $UBDPB = obj_{\text{PBP}}(y^{(k)})$, $y^* = y^{(k)}$, $k^* = k$.
      2. If Problem (PBP$_h^k$) is infeasible for scenario $\hat{h}$, stop solving it for scenarios $\hat{h} + 1, ..., s$ and set $S^k = S^{k-1} \cup \{k\}$. Then, set $\mu_h^{(k)} = 0$ for $h = 1, ..., \hat{h} - 1$, and solve the decomposed feasibility subproblem (FP$_h^k$) for $h = \hat{h}, ..., s$ and obtain the corresponding Lagrange multipliers $\mu_h^{(k)}$. Add a feasibility cut to Problem (RMP$^k$) according to $\mu_1^{(k)}, ..., \mu_s^{(k)}$.
      3. If $T^k = \emptyset$, solve the feasibility relaxed master problem (FRMP$^k$); otherwise, solve Problem (RMP$^k$). In the latter case, set $LBD$ to the optimal objective value of Problem (RMP$^k$) if Problem (RMP$^k$) is feasible. In both cases, set $y^{(k+1)}$ to the $y$ value at the solution of either problem.
    until $LBD \geq UBDPB$ or (Problem (RMP$^k$) or Problem (FRMP$^k$) is infeasible).
  end if
  if $UBDPB < UBD - \varepsilon$ then
    1. Solve the decomposed primal subproblem (PP$_h^*$) (i.e., for $y = y^*$) to $\varepsilon_h$-optimality for each scenario $h = 1, ..., s$ sequentially. Set $U^l = U^{l-1} \cup \{k^*\}$. If Problem (PP$_h^*$)
```

```
            has optimum $x_h^*$ for all the scenarios and $obj_{\mathrm{PP}}(y^*) = \sum_{h=1}^{s} obj_{\mathrm{PP}_h}(y^*) < UBD$,
            update $UBD = obj_{\mathrm{PP}}(y^*)$ and set $y_p^* = y^*$, $x_{p,h}^* = x_h^*$ for $h = 1, ..., s$.
        2. If $T^k \setminus U^l = \emptyset$, set $UBDPB = +\infty$.
        3. If $T^k \setminus U^l \neq \emptyset$, pick $i \in T^k \setminus U^l$ such that $obj_{\mathrm{PBP}}(y^{(i)}) = \min_{j \in T^k \setminus U^l} \{obj_{\mathrm{PBP}}(y^{(j)})\}$.
            Update $UBDPB = obj_{\mathrm{PBP}}(y^{(i)})$, $y^* = y^{(i)}$, $k^* = i$. Set $l = l + 1$.
    end if
  until $UBDPB \geq UBD - \varepsilon$ and (Problem (RMP$^k$) or Problem (FRMP$^k$) is infeasible or
  $LBD \geq UBD - \varepsilon$).
  An $\varepsilon$-optimal solution of Problem (P) is given by $(y_p^*, x_{p,1}^*, ..., x_{p,s}^*)$ or Problem (P) is
  infeasible.
```

4.2 Finite Convergence

Assumption 4.1. *Compared to Problem (PP$_h^l$), Problems (PBP$_h^k$) and (FP$_h^k$) (which are convex programs) and Problems (RMP^k) and (FRMP^k) (which are MILPs) can be solved with a much tighter tolerance, which is then negligible for the discussion of the ε-optimal solution of the NGBD algorithm.*

Assumption 4.2. *The optimal objective value of a problem returned by a global optimizer is greater than or equal to the real optimal objective value.*

Remark 4.1 *If a solution of Problem (PP^l) consists of ε_h-optimal solutions of Problem (PP$_h^l$) for $h = 1, ..., s$, then this solution is an ε-optimal solution of Problem (PP^l) when $\sum_{h=1}^{s} \varepsilon_h \leq \varepsilon$.*

Remark 4.2 *Notice that UBDPB is neither the upper bound, nor the lower bound for Problem (P). UBDPB has two functions in the algorithm. One is to control the "inner loop" of the algorithm (which is a GBD-like procedure). The other is to prevent solving Problem (PP) for any integer realization that will not lead to a global solution of Problem (P), and this is explained in Lemma 4.1.*

Remark 4.3 *Since an ε-optimal solution of Problem (PP^l) is used to update UBD, Remark 4.1 implies that the difference between UBD and the real upper bound is at most ε. So the* *satisfaction of the condition, $LBD < UBD - \varepsilon$, guarantees the gap between LBD and the real upper bound as expected. Similarly, the satisfaction of $UBDPB < UBD - \varepsilon$ guarantees the gap between $UBDPB$ and the real upper bound as expected.*

Lemma 4.1 *If the NGBD algorithm terminates finitely with a feasible solution of Problem (P), this feasible solution is an ε-optimal solution of Problem (P).*

Proof. Notice the algorithm terminates with "$UBDPB \geq UBD - \varepsilon$ and (Problem (RMP^k) or Problem (FRMP^k) is infeasible" or "$LBD \geq UBD - \varepsilon$)". First, it is demonstrated that this termination condition ensures that an integer realization which leads to an ε-optimal solution of Problem (P) has been visited by Problem (PBP). Second, it is demonstrated that if one such integer realization has been visited by Problem (PBP), the termination condition ensures that $y = y_p^*$ is one such integer realization and UBD is an ε-optimal objective value of Problem (P).

Consider the case in which Problem (RMP^k) or (FRMP^k) is infeasible. Since Problem (P) is feasible, Problem (FRMP^k) cannot be infeasible and the infeasibility of Problem (RMP^k) implies that all the feasible integer realizations have been visited by Problem (PBP), so any integer realization leading to an ε-optimal solution of Problem (P) has been visited by Problem (PBP).

Consider the case in which $LBD \geq UBD - \varepsilon$. Denote the real optimal objective value of the original problem (P) by $\widehat{obj}_{\text{P}}$. Denote the real optimal objective value of Problem (PP) for $y = y_p^*$ by $\widehat{obj}^*_{\text{PP}}$ and the one returned by the solver by obj^*_{PP}. Obviously,

$$\widehat{obj}^*_{\text{PP}} \geq \widehat{obj}_{\text{P}}. \tag{22}$$

According to Assumption 4.2,

$$UBD = obj^*_{\text{PP}} \geq \widehat{obj}^*_{\text{PP}}. \tag{23}$$

From (22) and (23),

$$UBD \geq \widehat{obj}_{\text{P}}. \tag{24}$$

Assume any integer realization that leads to an ε-optimal solution of Problem (P) has not been visited by Problem (PBP), then any such integer realization has not been excluded by the canonical integer cuts in Problem (RMP^k). According to Corollary 3.2,

$$\widehat{obj}_{\mathrm{P}} \geq LBD \geq UBD - \varepsilon. \tag{25}$$

Inequalities (24) and (25) imply that $y = y_p^*$ obtained at the termination of the algorithm leads to an ε-optimal solution of Problem (P) and this integer realization has been visited by Problem (PBP), which contradicts the assumption. Therefore, in the case in which $LBD \geq UBD - \varepsilon$, at least one integer realization that leads to an ε-optimal solution of Problem (P) has been visited by Problem (PBP) as well.

Finally, the algorithm ensures that $UBDPB$ equals to the minimum optimal objective of Problem (PBP) for those integer realizations that have been visited by Problem (PBP) but not by Problem (PP) (and $UBDPB = +\infty$ if no such integer realizations exist). Then at the termination when $UBDPB \geq UBD - \varepsilon$ always holds, such integer realizations cannot lead to a global optimal solution of Problem (P) due to Proposition 3.5. Therefore, an integer realization that leads to an ε-optimal solution of Problem (P) has been visited by Problem (PP), which has been recorded by $y = y_p^*$ and $UBD = obj_{\mathrm{PP}}^*$ is an ε-optimal objective value of Problem (P). □

Theorem 4.1 *If all the subproblems can be solved to ε-optimality in a finite number of steps, then the NGBD algorithm terminates in a finite number of steps with an ε-optimal solution of Problem (P) or an indication that Problem (P) is infeasible.*

Proof. Notice that all the integer realizations are generated by solving Problem (RMP^k) or (FRMP^k) in the algorithm. According to Corollary 3.1, no integer realizations will be generated twice. Since the cardinality of set Y is finite by definition and all the subproblems are terminated in finite number of steps, the algorithm terminates in a finite number of steps.

Lemma 4.1 shows that if Problem (P) is feasible, the algorithm terminates with its ε-optimal solution. If Problem (P) is infeasible, the algorithm terminates with $UBD = +\infty$

because UBD can only be updated with an ε-optimal solution of Problem (PP), which is infeasible for any integer realization in Y (and therefore UBD is never updated). □

5 Case Studies

5.1 Implementation

All the cases study problems were solved on a computer allocated a single 2.83 GHz CPU, 2 GB memory and running Linux Kernel. GAMS 23.4 [41] was used to formulate the problems, program the NGBD algorithm and interface the solvers for the subproblems. The NGBD method employed BARON 9.0.5 to obtain global solutions for the primal subproblems (PP_h), where CPLEX 12.1.0 was used as Linear Program (LP) solver and CONOPT 3 was used as the local NLP solver. The NGBD method itself employed CPLEX 12.1.0 to solve LP/MILP subproblems and CONOPT 3 to solve convex NLP subproblems. The feasibility problem (FP) minimized the 1-norm of the slack variable vector for all the case studies, using the standard smooth reformulation. The major purpose of the case study is to compare the computational efficiency of NGBD and two commercial global optimizers in GAMS 23.4, BARON and LINDOGLOBAL [42]. The relative and absolute termination criteria were set to be 10^{-3}. The solution times reported here are only the solver times reported by GAMS solvers.

Since this paper does not focus on scenario generation for stochastic programs, all the uncertain parameters in the case studies are assumed to obey normal distributions and are independent of each other. A naive sampling rule was used to generate scenarios for normally distributed uncertain parameters, which is detailed in the online supplementary material.

For all the case studies, convex underestimators for constructing the lower bounding problems were generated through McCormick relaxation [28] [43], i.e., through recursively applying rules for the relaxation of sums, products and univariate composition with the known convex and concave envelopes of the univariate intrinsic functions. In addition, auxiliary variables were introduced for the differentiability of the resulting underestimators [30].

The online supplementary material gives the convex and concave envelopes of the univariate intrinsic functions in the case studies and the relaxation rule for products.

5.2 Case Study Problems

The following three problems, which can be formulated into MINLPs in the form of Problem (P), are studied in this paper:

Software Reliability Problem – This problem is to find the optimal software structure for maximum software reliability while ensuring that expenditures remain within budget. The deterministic version of the problem was initially presented in [44] and then reformulated in [45] to avoid nonlinear functions of integer variables. In this paper the problem is revised into a two-stage stochastic problem which explicitly addresses the uncertainty in the reliabilities of three software modules and maximizes an expected reliability. The resulting problem has 8 binary variables, $8s$ continuous variables (where s denotes the total number of scenarios addressed in the problem) and 3 uncertain parameters.

Pump Network Configuration Problem – This problem is to find the optimal configuration of a centrifugal pump network for minimum annualized cost that achieves a prespecified pressure rise based on a given total flow rate. The deterministic version of the problem was initially presented in [46] and then updated in [22] with a set of additional linear constraints for tighter relaxation in global optimization. In this paper, the problem is further reformulated to reduce the number of nonlinear functions while exhibiting the structure of Problem (P). Then it is revised into a two-stage stochastic problem which explicitly addresses the uncertainty in the pump performance models and minimizes an expected annualized cost. The resulting problem has 18 binary variables, $38s$ continuous variables and 3 uncertain parameters.

Sarawak Gas Production Subsystem Problem – This problem comes from a real industrial system, the Sarawak Gas Production System (SGPS) [47]. In [48], optimal operation of a subsystem of the SGPS is studied. This problem is extended in this paper to a deterministic integrated design and operation problem which maximizes the net present value of the subsystem while satisfying the demand constraints at the end node. This deterministic problem is then revised into a two-stage stochastic problem which maximizes the expected net present value and explicitly addresses the uncertainty in the system, including product demand, product price and the pressure-flow relationship in a pipeline. The resulting problem has 20 binary variables, $110s$ continuous variables and 3 uncertain parameters.

Details of the above problems, including their GAMS models, can be found in the online supplementary material. The problems were solved with each uncertain parameter sampled for 1, 3, 5, 7, 9 and 11 values, which lead to stochastic programs with 1, 27, 125, 343, 729 and 1331 scenarios for all these problems (since each of them contains 3 uncertain parameters).

5.3 Results and Discussion

The results for the case study problems with different numbers of scenarios are summarized in Tables 1, 2 and 3. It can be seen that if only one scenario is addressed, BARON is faster than NGBD for the first two problems and LINDOGLOBAL is faster than NGBD for the pump network configuration problem. However, the solver time with BARON increases rapidly with the number of scenarios, and BARON did not return a solution within 10,000 seconds when no less than 125 scenarios were addressed for the pump network configuration problem and no less than 27 scenarios were addressed for the SGPS problem. LINDOGLOBAL was not able to solve any of the problems with no less than 27 scenarios. (Also notice that LINDOGLOBAL is restricted to problems with up to 3,000 variables and 2,000 constraints.) On the other hand, the solver time with NGBD increases slowly with the number of scenarios. NGBD solved the largest problem in the case study, the SGPS problem with 1331 scenarios and almost 150,000 variables, within 80 minutes of solver time. And NGBD can solve all the case study problems with much more scenarios within reasonable time, according to the trend shown by the computational results.

Obviously, the time for obtaining global optima for Problems (PP$_h$) dominated the total solver time in NGBD. By exploiting UBD_h, which can be calculated according to the solutions of the subproblems solved in the previous iterations (as defined in Proposition 3.6), the solution time for Problem (PP$_h$) can be reduced (so that the total solver time for Problem (P) can be reduced). For the software reliability problem, the time for solving Problem (PP_h) was reduced by up to 60% when UBD_h was used to accelerate the solution, while for the other two problems the time was reduced by about 30% and 10%, respectively.

Tables 1, 2 and 3 also show the numbers of integer realizations visited by Problem (PBP_h) and Problem (PP_h), which are also the numbers of iterations in the "inner loop" and in the "outer loop" of the NGBD algorithm, respectively. Also, the number of integer realizations visited by Problem (PBP_h) indicates the size of the largest relaxed master problem solved in NGBD (since the number of cuts in Problem (RMP^k) is twice of the number of such integer realizations). Notice that NGBD avoided visiting most of the integer realizations in Y for the case study problems. In the software reliability problem, only about 15% of integer realizations in Y were visited by Problem (PBP_h). For the other two problems such data are less than 10% and less than 15%. In addition, the number of the visited integer realizations does not change significantly with the number of scnerios, i.e, the sizes of the relaxed master problems can be deemed to be independent of the number of scenarios. Also notice that not all such integer realizations are visited by Problem (PP_h); in many cases the ratio of the integer realizations visited by Problem (PP_h) to those visited by Problem (PBP_h) is only about 1/4. This is because in NGBD the solution of Problem (PP_h) is postponed as much as possible and the solution of Problem (PBP_h) helps to eliminate some integer realizations that will never lead to a global optimum (as indicated by Proposition 3.5). As a result, NGBD is much more efficient than a naive integer enumeration in which Problems (PP_h) are solved for all the integer realizations in Y (which would also solve subproblems whose sizes do not grow with the number of scenarios).

Finally, the tables show UBD and LBD at the NGBD termination for all the cases. UBD stands for an upper bound as well as an ε-optimal objective value of Problem (P). LBD stands for the lower bound of the Problem (P) excluding all the visited integer realizations, so LBD can be significantly larger than UBD at the termination and $LBD = +\infty$ implies that all the feasible integer realizations have been visited. The convergence property of NGBD is also demonstrated through Figures 2, 3 and 4, which show the values of UBD and LBD over the iterations of NGBD for the three case study problems with 1331 scenarios, respectively.

Table 1 Results for the software reliability problem (Unit for solver time: second)

Number of Scenarios	1	27	125	343	729	1331
Number of continuous variables	8	216	1000	2744	5832	10648
Number of binary variables	8	8	8	8	8	8
Total time with BARON	0.2	1.5	45.2	240.0	1650.8	6784.5
Total time with LINDOGLOBAL	0.4	$*^a$	$*$	$*$	$*$	$*$
Total time with NGBD	0.2	4.2	24.6	65.2	139.8	260.7
Detailed results for NGBD						
Time for PBP$_h$	0.0	0.9	4.8	13.3	27.5	54.9
Time for FP$_h$	0	0	0	0	0	0
Time for RMP&FRMP	0.0	0.0	0.1	0.1	0.1	0.1
Time for PP$_h$	0.1	3.3	19.7	51.8	112.2	205.8
Time for PP$_h$ (UBD_h not exploited)	0.2	6.7	46.9	133.7	271.3	501.1
UBD at termination (%)	-94.37	-93.16	-93.16	-93.16	-93.16	-93.16
LBD at termination (%)	-93.56	-93.19	-91.96	-91.96	-91.96	-91.96
Integer realizations visited by PBP$_h$	9	10	13	13	13	13
Integer realizations visited by PP$_h$	5	6	9	9	9	9
Total integer realizations in Y	81	81	81	81	81	81

[a] Problem size exceeds the limit of LINDOGLOBAL or the solver terminated with failure.

Table 2 Results for the pump network configuration problem (Unit for solver time: second)

Number of Scenarios	1	27	125	343	729	1331
Number of continuous variables	38	1026	4750	13034	27702	50578
Number of binary variables	18	18	18	18	18	18
Total time with BARON	0.53	28.9	- [a]	-	-	-
Total time with NGBD	7.7	60.9	328.8	754.1	1497.0	2794.8
Total time with LINDOGLOBAL	5.7	$*^b$	$*$	$*$	$*$	$*$
Detailed results for NGBD						
Time for PBP$_h$	0.3	1.9	2.8	9.6	39.0	73.8
Time for FP$_h$	0.1	3.8	7.1	21.5	111.2	230.3
Time for RMP&FRMP	1.8	1.0	0.7	0.7	1.1	1.1
Time for PP$_h$	5.4	54.2	318.3	722.4	1345.7	2489.6
Time for PP$_h$ (UBD_h not exploited)	5.5	81.1	351.3	943.8	1978.3	3470.0
UBD at termination (FIM)	128.9	136.6	136.3	145.3	145.3	145.3
LBD at termination (FIM)	$+\infty$ [c]	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$
Integer realizations visited by PBP$_h$	100	72	77	75	77	80
Integer realizations visited by PP$_h$	41	21	20	19	19	19
Total integer realizations in Y	1024	1024	1024	1024	1024	1024

[a] No solution returned within 10,000 seconds.

[b] Problem size exceeds the limit of LINDOGLOBAL or the solver terminated with failure.

[c] Represented by a big number (10^{10}) in NGBD. It indicates all feasible integer realizations have been visited by PBP$_h$.

Table 3 Results for the SGPS problem (Unit for solver time: second)

Number of Scenarios	1	27	125	343	729	1331
Number of continuous variables	110	2970	13750	37730	80190	146410
Number of binary variables	20	20	20	20	20	20
Total time with BARON	123.2	-[a]	-	-	-	-
Total time with LINDOGLOBAL	$*$[b]	$*$	$*$	$*$	$*$	$*$
Total time with NGBD	0.6	78.7	372.0	1081.2	2253.1	4234.8
Detailed results for NGBD						
Time for PBP$_h$	0.1	5.5	31.3	81.1	172.7	259.4
Time for FP$_h$	0.0	0.0	0.5	1.2	2.7	4.0
Time for RMP&FRMP	0.1	0.8	0.8	0.9	0.8	0.6
Time for PP$_h$	0.5	72.4	339.3	998.1	2076.9	3970.8
Time for PP$_h$ (UBD_h not exploited)	0.5	81.4	368.4	1370.3	2460.9	4490.0
UBD (Billion \$)	-7.209	-7.189	-7.187	-7.188	-7.188	-7.188
LBD (Billion \$)	-7.209	-7.189	-7.189	-7.189	-7.189	-7.189
Integer realizations visited by PBP$_h$	14	70	70	71	70	68
Integer realizations visited by PP$_h$	1	16	16	16	16	16
Total integer realizations in Y	512	512	512	512	512	512

[a] No solution returned within 10,000 seconds.

[b] Problem size exceeds the limit of LINDOGLOBAL or the solver terminated with failure.

6 Conclusions

This paper extends traditional GBD to solving rigorously stochastic separable MINLP problems in the form of Problem (P). A lower bounding problem is introduced for which GBD iterations can be applied rigorously, and the solution of the lower bounding problem yields both valid lower bounds for the problem and the integer realizations that can be used to construct valid upper bounding problems. The resulting method, called NGBD, solves a sequence of subproblems whose sizes are independent of the number of scenarios addressed in the problem. This method terminates finitely with an ε-optimal solution or an indication of the infeasibility of the problem. The dramatic computational advantage of NGBD over a state-of-the-art branch-and-reduce global optimization solver, BARON, is demonstrated by the results of the case studies, in which a highly nonconvex MINLP with almost 150,000 variables was solved with NGBD within 80 minutes of solver time. The results also show that the time for solving the primal subproblems dominates the total solution time, and this time can be reduced by exploiting the solutions of the subproblems solved in the previous iterations. Since most of the subproblems in an iteration can be solved without exchanging

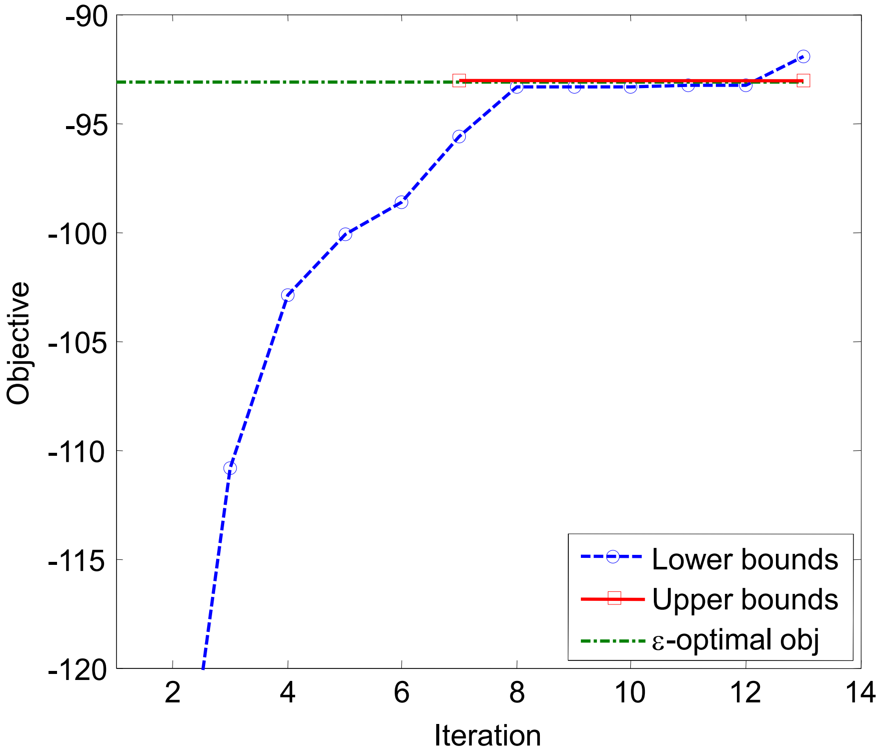

Figure 2 Evolution of *UBD* and *LBD* over NGBD iterations for the software reliability problem with 1331 scenarios.

information among them, the performance of NGBD can be further improved by exploitation of a parallel computation architecture.

The extension of the proposed NGBD method to multistage problems will be interesting future work. Notice that NGBD is not advantageous if a multistage stochastic program is formulated into a special two-stage problem with nonanticipativity constraints (that equate the decisions in different scenarios and with the same uncertainty realizations in all previous stages) [12], because then the problem cannot be decomposed for different scenarios by fixing the first-stage decisions (due to the presence of nonanticipativity constraints). On the other hand, NGBD can be applied to a K-stage problem in a recursive way provided the decision variables in stages $1,...,K-1$ are all integers, because such a K-stage problem can be treated as $K-1$ nested two-stage problems in the form of Problem (P). A similar idea has been used to solve multistage linear programs with Benders decomposition in the

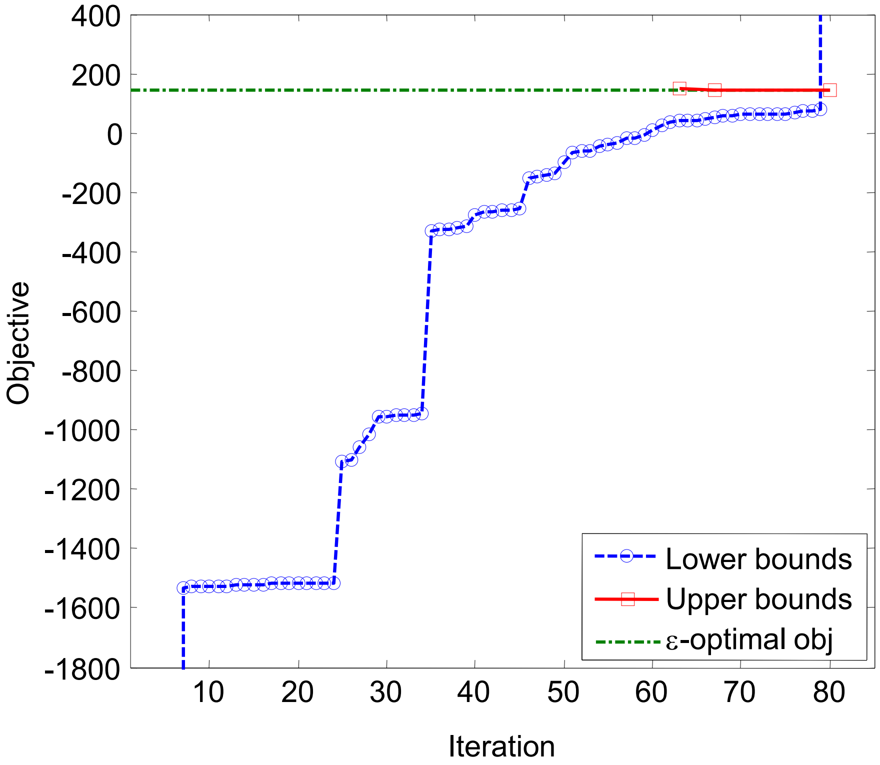

Figure 3 Evolution of *UBD* and *LBD* over NGBD iterations for the pump network configuration problem with 1331 scenarios.

stochastic programming literature (e.g., [49]). Howerver, if some of the decision variables in stages $1,\ldots,K-1$ are continuous, the K-stage problem will contain two-stage problems with continuous complicating variables, which cannot be represented in the form of Problem (P) and solved by the current NGBD method.

References

1. I. E. Grossmann. Review of nonlinear mixed-integer and disjunctive programming techniques. *Optimization and Engineering*, 3:227–252, 2002.
2. C. A. Floudas. *Nonlinear and Mixed-Integer Optimization: Fundamentals and Applications*. Oxford University Press, 1995.
3. L. T. Biegler and I. E. Grossmann. Retrospective on optimization. *Computers and Chemical Engineering*, 28:1169–1192, 2004.

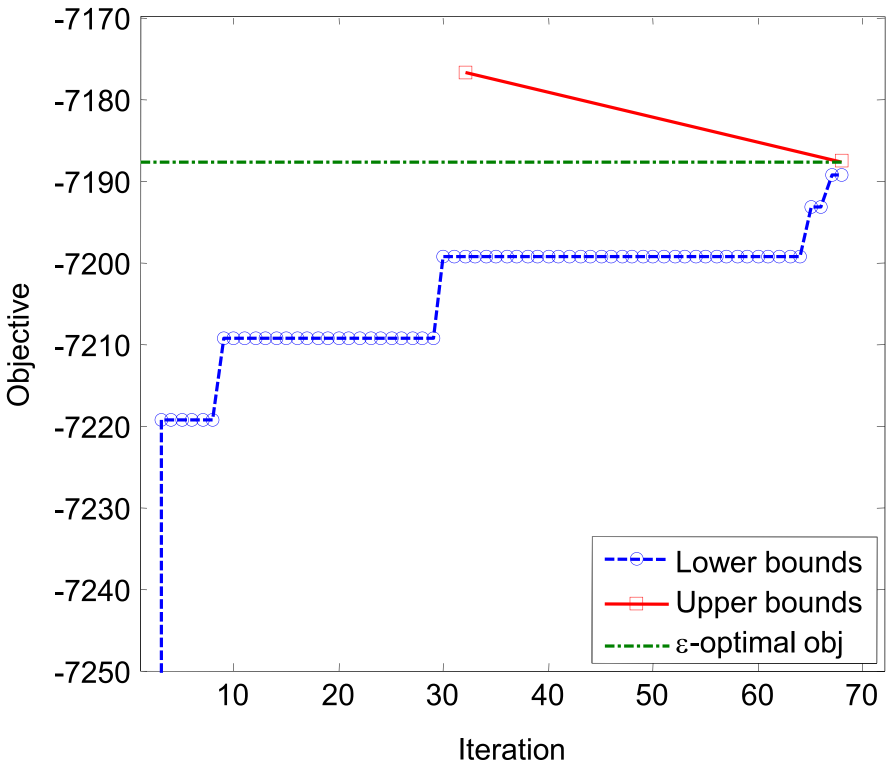

Figure 4 Evolution of *UBD* and *LBD* over NGBD iterations for the SGPS problem with 1331 scenarios.

4. N.V. Sahinidis. Optimization under uncertainty: state-of-the-art and opportunities. *Computers and Chemical Engineering*, 28:971–983, 2004.

5. J. R. Birge and F. Louveaux. *Introduction to Stochastic Programming*. Springer, New York, 1997.

6. X. Li, E. Armagan, A. Tomasgard, and P. I. Barton. Stochastic pooling problem for natural gas production network design and operation under uncertainty. *AIChE Journal*, 2010. DOI: 10.1002/aic.12419.

7. B. Tarhan, I. E. Grossmann, and V. Goel. Stochastic programming approach for the planning of offshore oil or gas field infrastructure under decision-dependent uncertainty. *Industrial and Engineering Chemistry Research*, 48:3078–3097, 2009.

8. R. Karuppiah and I. E. Grossmann. Global optimization of multiscenario mixed integer nonlinear programming models arising in the synthesis of integrated water networks under uncertainty. *Computers and Chemical Engineering*, 32:145–160, 2008.

9. S. Rebennack, J. Kallrath, and P. M. Pardalos. Optimal storage design for a multi-product plant: A non-convex MINLP formulation. *Computers and Chemical Engineering*, 35:255–271, 2011.

10. J. Acevedo and E. N. Pistikopoulos. Stochastic optimization based algorithms for process synthesis under uncertainty. *Computers and Chemical Engineering*, 22:647–671, 1998.

11. S. Ahmed, A. J. King, and G. Parija. A multi-stage stochastic integer programming approach for capacity expansion under uncertainty. *Journal of Global Optimization*, 26:3–24, 2003.

12. A. Shapiro, D. Dentcheva, and A. Ruszczynski. *Lectures on Stochastic Programming: Modeling and Theory*. SIAM, Philadelphia, 2009.

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

14. R. M. Van Slyke and R. Wets. *L*-shaped linear programs with applications to optimal control and stochastic programming. *SIAM Journal on Applied Mathematics*, 17(4):638–663, 1969.

15. A. M. Geoffrion. Generalized Benders decomposition. *Journal of Optimization Theory and Applications*, 10(4):237–260, 1972.

16. C. C. Carøe and R. Schultz. Dual decomposition in stochastic integer programming. *Operations Research Letters*, 24:37–45, 1999.

17. M. L. Fisher. The Lagrangian relaxation method for solving integer programming problems. *Management Science*, 27:1–18, 1981.

18. M. L. Fisher. An applications oriented guide to Lagrangian relaxation. *Interfaces*, 15(2):10–21, 1985.

19. R. Karuppiah and I. E. Grossmann. A Lagrangean based branch-and-cut algorithm for global optimization of nonconvex mixed-integer nonlinear programs with decomposable structures. *Journal of Global Optimization*, 41:163–186, 2008.

20. A. Khajavirad and J. J. Michalek. A deterministic Lagrangian-based global optimization approach for quasiseparable nonconvex mixed-integer nonlinear programs. *Journal of Mechanical Design*, 131(5):051009–1–051009–8, 2009.

21. M. Tawarmalani and N. V. Sahinidis. Global optimization of mixed-integer nonlinear programs: A theoretical and computational study. *Mathematical Programming*, 99:563–591, 2004.

22. C. S. Adjiman, I. P. Androulakis, and C. A. Floudas. Global optimization of mixed-integer nonlinear problems. *AIChE Journal*, 46(9):1769–1797, 2000.

23. P. Kesavan, R. J. Allgor, E. P. Gatzke, and P. I. Barton. Outer approximation algorithms for separable nonconvex mixed-integer nonlinear programs. *Mathematical Programming, Series A*, 100:517–535, 2004.

24. M. Duran and I. E. Grossmann. An outer-approximation algorithm for a class of mixed nonlinear programs. *Mathematical Programming*, 66:327–349, 1986.

25. R. Fletcher and S. Leyffer. Solving mixed integer nonlinear programs by outer approximation. *Mathematical Programming*, 66:327–349, 1994.

26. A. M. Geoffrion. Elements of large-scale mathematical programming: Part I: Concepts. *Management Science*, 16(11):652–675, 1970.

27. A. M. Geoffrion. Elements of large-scale mathematical programming: Part II: Synthesis of algorithms and bibliography. *Management Science*, 16(11):652–675, 1970.

28. G. P. McCormick. Computability of global solutions to factorable nonconvex programs: Part I - Convex underestimating problems. *Mathematical Programming*, 10:147–175, 1976.

29. C. S. Adjiman, S. Dallwig, C. A. Floudas, and A. Neumaier. A global optimization method, α-BB, for general twice-differentiable constrained NLPs – I. Theoretical advances. *Computers and Chemical Engineering*, 22(9):1137–1158, 1998.

30. E. P. Gatzke, J. E. Tolsma, and P. I. Barton. Construction of convex relaxations using automated code generation technique. *Optimization and Engineering*, 3:305–326, 2002.

31. S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, Cambridge, UK, 2004.

32. N. V. Sahinidis and M. Tawarmalani. *BARON 9.0.4: Global Optimization of Mixed-Integer Nonlinear Programs,* User's Manual, 2010.

33. ARKI Consulting and Development. http://www.gams.com/docs/conopt3.pdf.

34. P. E. Gill, W. Murray, and M. A. Saunders. SNOPT: an SQP algorithm for large-scale constrained optimization. *SIAM Review*, 47:99–131, 2005.

35. IBM. IBM ILOG CPLEX: High-performance mathematical programming engine. http://www-01.ibm.com/software/integration/optimization/cplex/.

36. C. Lemaréchal and C. Sagastizábal. Variable metric bundle methods: From conceptual to implementable forms. *Mathematical Programming*, 76:393–410, 1997.

37. E. Balas and R. Jeroslow. Canonical cuts on the unit hypercube. *SIAM Journal on Applied Mathematics*, 23(1):61–69, 1972.

38. D. P. Bertsekas. *Nonlinear Programming*. Athena Scientific, Cambridge, MA, 2nd edition, 1999.

39. A. M. Geoffrion. Duality in nonlinear programming: a simplified applications-oriented development. *SIAM Review*, 13(1):1–37, 1971.

40. M. S. Bazaraa, H. D. Sherali, and C. M. Shetty. *Nonlinear Programming: Theory and Algorithms*. John Wiley & Sons, Inc, New York, 2nd edition, 1993.

41. GAMS. General Algebraic and Modeling System. Available at http://www.gams.com/.

42. LindoSystems, Inc. LINDOGlobal User's Manual. http://www.gams.com/dd/docs/solvers/lindoglobal.pdf.

43. A. Mitsos, B. Chachuat, and P. I. Barton. McCormick-based relaxations of algorithms. *SIAM Journal on Optimization*, 20(2):573–601, 2009.

44. O. Berman and N. Ashrafi. Optimization models for reliability of modular software systems. *IEEE Transactions on Software Engineering*, 19(11):1119–1123, 1993.

45. C. A. Floudas, P. M. Pardalos, C. S. Adjiman, W. R. Esposito, Z. Gumus, S. T. Harding, J. L. Klepeis, C. A. Meyer, and C. A. Schweiger. *Handbook of Test Problems for Local and Global Optimization*. Kluwer Academic Publishers, 1999.

46. T. Westerlund, F. Pettersson, and I. E. Grossmann. Optimization of pump configurations as a MINLP problem. *Computers and Chemical Engineering*, 18(9):845–858, 1994.

47. A. Selot, L. K. Kuok, M. Robinson, T. L. Mason, and P. I. Barton. A short-term operational planning model for natural gas production systems. *AIChE Journal*, 54(2):495–515, 2008.

48. A. Selot. *Short-term Supply Chain Management in Upstream Natural Gas Systems*. PhD thesis, Massachusetts Institute of Technology, 2009.

49. J. R. Birge. Decomposition and partitioning methods for multistage stochastic linear programs. *Operations Research*, 33(5):989–1007, 1985.