49th IEEE Conference on Decision and Control
December 15-17, 2010
Hilton Atlanta Hotel, Atlanta, GA, USA

Decomposition Strategy for Natural Gas Production Network Design Under Uncertainty

Xiang Li, Asgeir Tomasgard and Paul I. Barton

Abstract— **The use of natural gas for power generation has been rising rapidly in the past two decades [1]. To ensure the security of supply of gas to the market and meet strict specifications on gas quality (e.g., sulfur content), natural gas production network design must address uncertainty explicitly as well as tracking the quality of each gas flow in the entire system. This leads to the stochastic pooling problem [2], which is a (potentially large-scale) nonconvex mixed-integer nonlinear program (MINLP). This paper presents a rigorous, duality-based decomposition strategy to solve the stochastic pooling problem, which guarantees finding an ε-optimal solution of the problem with a finite number of iterations. A case study involving a gas production network demonstrates the dramatic computational advantages of the decomposition method over a state-of-the-art global optimization method. The proposed method can be extended to tackle more general nonconvex MINLP problems, which may occur in the design of integrated energy systems involving fuel production, power generation and electricity transmission [3].**

I. INTRODUCTION

Natural gas is an important fuel for electricity generation worldwide, because it is more efficient and less carbon-intensive than other fossil fuels [4]. In the past two decades, the use of natural gas for electricity generation has been rising rapidly, and this trend is expected to continue in the next two decades [1] [4]. Traditionally, electric power systems are planned and operated without considering other energy subsystems, which may not lead to the best performance for the overall energy system including fuel production, power generation and electricity transmission subsystems. This has motivated research, in both electrical engineering (e.g., [3] [5] [6]) and chemical engineering (e.g., [7]) communities, on modeling and optimization of integrated energy systems. This paper focuses on solution algorithms for difficult optimization problems that arise in the design of natural gas production networks (that are part of integrated energy systems), and this technique can also be applied to the design of integrated energy systems that include natural gas production subsystems.

There are two major challenges in natural gas production network design. One is to track the qualities, or the compositions, of the gas flows throughout the entire system. This is because the qualities of the raw gas flows from different reservoirs can vary over large ranges, but these gas flows are usually sent through several mixing and splitting units, with little further processing, to product terminals, at which they must satisfy strict quality specifications. The other is to address the large uncertainty in different parts of the system (e.g., capacity and quality of reservoirs, customer demands, etc.). Li et al. addressed these two difficult issues with a stochastic pooling problem formulation [2], where the qualities of the gas flows are modeled with bilinear functions (which are nonconvex) and the uncertainty in the natural gas production networks is represented by a finite number of scenarios in the stochastic programming formulation.

In [2], the stochastic pooling problem was solved by a state-of-the-art global optimization solver, BARON [8] and it was shown that BARON was not suitable for the problems with large numbers of scenarios. This paper is devoted to a more efficient solution method for the stochastic pooling problem in the following form:

$$\begin{aligned}
\min_{\substack{y,x_1,\dots,x_s,\\ q_1,\dots,q_s,u_1,\dots,u_s}} \quad & c_1^{\mathrm{T}} y + \sum_{h=1}^{s} (c_{2,h}^{\mathrm{T}} x_h + c_{3,h}^{\mathrm{T}} q_h + c_{4,h}^{\mathrm{T}} u_h) \\
\text{s.t.} \quad & u_{h,l,t} = x_{h,l} q_{h,t},\ \forall (l,t) \in \Omega \subset \{1,\dots,n_x\} \times \{1,\dots,n_q\} \\
& A_{1,h} y + A_{2,h} x_h + A_{3,h} q_h + A_{4,h} u_h \leq b_h, \\
& (x_h, q_h, u_h) \in \Pi_h \subset \mathbb{R}^{n_x} \times \mathbb{R}^{n_q} \times \mathbb{R}^{n_u},\ y \in Y \subset \{0,1\}^{n_y}, \\
& \forall h \in \{1,\dots,s\},
\end{aligned} \tag{P}$$

where the index $h \in \{1,...,s\}$ indicates the different scenarios of uncertainty realization, Π_h is a nonempty compact convex polyhedral set for each scenario h, and Y is a set of binary variables.

The stochastic pooling problem is a potentially large-scale nonconvex MINLP problem. MINLP problems are typically solved with branch-and-bound strategies, such as branch-and-reduce [8], SMIN-αBB and GMIN-αBB [9], or decomposition strategies, such as outer approximation [10] [11] [12] and generalized Benders decomposition (GBD) [13]. In the context of scenario-based stochastic programming, the duality-based decomposition strategies, such as GBD, can naturally decompose the problem for each scenario, which is an overwhelming advantage over other solution strategies. However, GBD is restricted to specific convex programs and it cannot be applied to nonconvex MINLPs directly. In this paper, a rigorous, duality-based decomposition strategy which is inspired by GBD, is developed for the stochastic pooling problem. The proposed method guarantees finding an

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.

X. Li is with Department of Chemical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139, USA xiangli@mit.edu

A. Tomasgard is with Department of Industrial Economics and Technology Management, Norwegian University of Science and Technology, Trondheim, 7491, Norway asgeir.tomasgard@iot.ntnu.no

P. I. Barton is with Department of Chemical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139, USA pib@mit.edu

ε-optimal solution with a finite number of iterations. This paper does not give all the details of the method because of the limited space, but it presents the general idea and most important aspects of the strategy, so that readers can understand the essence of the method.

The remaining part of the paper is organized as follows: Section II gives an overview of the decomposition strategy and Section III discusses the decomposed subproblems. Then, Section IV details the decomposition algorithm. Section V presents the results of a case study that demonstrate the dramatic computational advantage of the proposed method over a state-of-the-art global optimization method. The paper ends with conclusions and discussions on future work in Section VI.

II. THE DECOMPOSITION STRATEGY

The decomposition strategy is developed based on the framework of concepts presented by Geoffrion for the design of large-scale mathematical programming techniques [14]. This framework includes two groups of concepts: problem manipulations and solution strategies. Problem manipulations, such as **convexification**, **projection** and **dualization** (used in this paper), 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, such as **relaxation** and **restriction** (used in this paper), reduce the master problem to a related sequence of simpler subproblems.

A. Convexification - The Lower Bounding Problem

Since Problem (P) is separable in the continuous and the integer variables, the continuous and integer feasible regions can be individually characterized [10]. So it suffices to convexify and underestimate the bilinear functions in Problem (P) to yield a lower bounding problem. This can be done by replacing the bilinear functions in Problem (P) with their convex and concave envelopes [15], which leads to the following mixed-integer linear program (MILP):

$$\begin{aligned} \min_{\substack{y,x_1,\dots,x_s,\\ q_1,\dots,q_s,u_1,\dots,u_s}} \quad & c_1^{\mathrm{T}} y + \sum_{h=1}^{s} (c_{2,h}^{\mathrm{T}} x_h + c_{3,h}^{\mathrm{T}} q_h + c_{4,h}^{\mathrm{T}} u_h) \\ \text{s.t.} \quad & A_{1,h} y + A_{2,h} x_h + A_{3,h} q_h + A_{4,h} u_h \le b_h, \\ & (x_h, q_h, u_h) \in \hat{\Pi}_h, \quad y \in Y, \\ & \forall h \in \{1,\dots,s\}, \end{aligned} \tag{LBP}$$

where

$$\begin{aligned} \hat{\Pi}_h = \{ & (x_h,q_h,u_h) : (x_h,q_h,u_h) \in \Pi_h, \\ & u_{h,l,t} \ge x_{h,l}^{\mathrm{L}} q_{h,t} + x_{h,l} q_{h,t}^{\mathrm{L}} - x_{h,l}^{\mathrm{L}} q_{h,t}^{\mathrm{L}}, \\ & u_{h,l,t} \ge x_{h,l}^{\mathrm{U}} q_{h,t} + x_{h,l} q_{h,t}^{\mathrm{U}} - x_{h,l}^{\mathrm{U}} q_{h,t}^{\mathrm{U}}, \\ & u_{h,l,t} \le x_{h,l}^{\mathrm{U}} q_{h,t} + x_{h,l} q_{h,t}^{\mathrm{L}} - x_{h,l}^{\mathrm{U}} q_{h,t}^{\mathrm{L}}, \\ & u_{h,l,t} \le x_{h,l}^{\mathrm{L}} q_{h,t} + x_{h,l} q_{h,t}^{\mathrm{U}} - x_{h,l}^{\mathrm{L}} q_{h,t}^{\mathrm{U}}, \ \forall (l,t) \in \Omega \} \end{aligned}$$

is the intersection of set Π_h and the convex and concave envelopes for the bilinear functions, and $x_{h,l}^U$, $x_{h,l}^L$ denote the upper and lower bounds on $x_{h,l}$ and $q_{h,t}^U$, $q_{h,t}^L$ denote the upper and lower bounds on $q_{h,t}$. Obviously, $\hat{\Pi}_h$ is also a nonempty compact convex polyhedral set for each scenario h. It is not difficult to show that any feasible point of Problem (P) is also feasible for Problem (LBP), and the optimal objective value of Problem (LBP) represents a lower bound on the optimal objective value of Problem (P).

B. Projection/Dualization - The Master Problem

Problem (LBP) is a large-scale MILP when the number of scenarios addressed by the problem is large. According to the principle of projection explained in Geoffrion [13], Problem (LBP) can be projected from the feasible region of both the continuous and integer variables to the feasible region of the integer variables, and any subproblem for a fixed integer realization can be reformulated into its dual. Thus, Problem (LBP) can be transformed into the following form:

$$\begin{aligned} \min_{y,\eta} \quad & \eta \\ \text{s.t.} \quad & \eta \ge F(y,\lambda_1,\dots,\lambda_s), \quad \forall \lambda_1,\dots,\lambda_s \ge 0, \\ & G(y,\mu_1,\dots,\mu_s) \le 0, \quad \forall (\mu_1,\dots,\mu_s) \in M, \\ & y \in Y, \ \eta \in \mathbb{R}, \end{aligned} \tag{MP}$$

where

$$F(y,\lambda_1,\dots,\lambda_s) = \inf_{\substack{(x_h,q_h,u_h)\in\hat{\Pi}_h,\\ \forall h\in\{1,\dots,s\}}} [c_1^{\mathrm{T}} y + \sum_{h=1}^{s} (c_{2,h}^{\mathrm{T}} x_h + c_{3,h}^{\mathrm{T}} q_h + c_{4,h}^{\mathrm{T}} u_h) + \sum_{h=1}^{s} \lambda_h^{\mathrm{T}} g_h(y,x_h,q_h,u_h)],$$

$$G(y,\mu_1,\dots,\mu_s) = \inf_{\substack{(x_h,q_h,u_h)\in\hat{\Pi}_h,\\ \forall h\in\{1,\dots,s\}}} \sum_{h=1}^{s} \mu_h^{\mathrm{T}} g_h(y,x_h,q_h,u_h),$$

and

$$g_h(y,x_h,q_h,u_h) = A_{1,h} y + A_{2,h} x_h + A_{3,h} q_h + A_{4,h} u_h - b_h,$$

the multipliers

$$\lambda_h \in \mathbb{R}^m, \ \mu_h = (\mu_{h,1},\dots,\mu_{h,m}) \in \mathbb{R}^m, \quad \forall h \in \{1,\dots,s\},$$

and the set

$$M = \{(\mu_1,\dots,\mu_s) \in \mathbb{R}^{m\times s} : \mu_1,\dots,\mu_s \ge 0, \ \sum_{h=1}^{s}\sum_{i=1}^{m} \mu_{h,i} = 1\}.$$

The first set of constraints in Problem (MP) represents the optimality cuts, which restrict the optimal objective value of the problem according to strong duality for linear programs (LP). The second set of constraints represents the feasibility cuts, which characterize the feasible region of the problem in the projected space. The equivalence of Problems (MP) and (LBP) follows from Theorems 2.1, 2.2 and 2.3 in [13].

C. Relaxation/Restriction - Solution Strategies

The master problem (MP) is difficult to solve directly because of the uncountable number of constraints. Therefore, it is solved in this paper by solving a sequence of **Relaxed Master Problems** and **Primal Bounding Problems** which are much easier to solve. The primal bounding problem is constructed by restricting the integer variables to specific values in the lower bounding problem (LBP), whose solution yields a valid upper bound on the optimal objective value of Problem (LBP) (and therefore Problem (MP) as well). When the primal bounding problem is infeasible, a corresponding **Feasibility Problem** is solved, which yields valid information for the algorithm to proceed. Both the primal bounding and the feasibility problems can be further decomposed into subproblems for each scenario. The relaxed master problem is constructed by relaxing Problem (MP) so that the constraints are satisfied for only a finite number of multiplier values. The solution of the relaxed master problem yields a valid lower bound on the optimal objective value of Problem (MP).

On the other hand, Problem (LBP) or (MP) is a relaxation of the original problem (P), and a restriction of Problem (P), called the **Primal Problem**, is constructed by restricting the integer variables to specific values in Problem (P). The primal problem can also be further decomposed into subproblems for each scenario.

The proposed decomposition algorithm is implemented by solving the aforementioned subproblems iteratively until certain stopping criteria are met and a global optimum or infeasibility of Problem (P) is indicated. More details of the decomposed subproblems are given in the next section.

III. THE DECOMPOSED SUBPROBLEMS

A. Decomposed Primal Bounding Subproblem

When visiting the kth integer realization of $y = y^{(k)}$, Problem (LBP) is reduced to an LP, which can naturally be decomposed into s subproblems. For each scenario h, the subproblem is as follows:

$$\begin{aligned} obj_{\mathrm{PBP}_h}(y^{(k)}) = \min_{x_h,q_h,u_h} \; & c_{2,h}^{\mathrm{T}} x_h + c_{3,h}^{\mathrm{T}} q_h + c_{4,h}^{\mathrm{T}} u_h \\ \text{s.t.} \quad & A_{1,h} y^{(k)} + A_{2,h} x_h + A_{3,h} q_h + A_{4,h} u_h \le b_h, \\ & (x_h, q_h, u_h) \in \hat{\Pi}_h. \end{aligned} \qquad (\mathrm{PBP}_h)$$

The Lagrange multipliers for Problem (PBP_h) for $h = 1, \ldots, s$ are used to construct optimality cuts for the relaxed master problem (which will be discussed later).

B. Decomposed Feasibility Subproblem

If Problem (PBP_h) is infeasible for some $h \in \{1, \ldots, s\}$ when $y = y^{(k)}$, then the primal bounding problem is infeasible. In this case, a corresponding feasibility problem is solved, which minimizes the violation of the constraints by introducing additional slack variables. Again, the feasibility problem can be naturally decomposed into s subproblems. For each scenario h, the subproblem is as follows:

$$\begin{aligned} \min_{x_h,q_h,u_h,z_h} \quad & \sum_{i=1}^{m} z_{h,i} \\ \text{s.t.} \quad & A_{1,h} y^{(k)} + A_{2,h} x_h + A_{3,h} q_h + A_{4,h} u_h - b_h \le z_h, \\ & (x_h, q_h, u_h) \in \hat{\Pi}_h, \; z_h = (z_{h,1}, \ldots, z_{h,m}) \in \{z \in \mathbb{R}^m : z \ge 0\}. \end{aligned} \qquad (\mathrm{FP}_h)$$

It can be shown that the Lagrange multipliers for Problem (FP_h) for $h = 1, \ldots, s$ form valid multipliers to construct the feasibility cuts for the relaxed master problem. The proof is not given here due to space limitations.

C. Relaxed Master Problem

After solving the primal bounding subproblems or feasibility subproblems for a particular integer realization, a relaxed master problem will be solved to generate a new integer realization, which only contains a finite number of optimality and feasibility cuts that are constructed with the Lagrange multipliers for the primal bounding subproblems and feasibility subproblems solved at previous iterations. This problem is stated for iteration k as follows:

$$\begin{aligned} \min_{y,\eta} \quad & \eta \\ \text{s.t.} \quad & \eta \ge F(y, \lambda_1^{(j)}, \ldots, \lambda_s^{(j)}), \quad \forall j \in T^k, \\ & G(y, \mu_1^{(i)}, \ldots, \mu_s^{(i)}) \le 0, \quad \forall i \in S^k, \\ & \sum_{r \in \{r: y_r^{(p)} = 1\}} y_r - \sum_{r \in \{r: y_r^{(p)} = 0\}} y_r \le |\{r : y_r^{(p)} = 1\}| - 1, \\ & \quad \forall p \in T^k \cup S^k, \\ & y \in Y, \; \eta \in \mathbb{R}, \end{aligned} \qquad (\mathrm{RMP}^k)$$

where the index sets T^k and S^k contain the indices of the previous iterations in which the primal bounding problem is feasible and infeasible, respectively. The additional constraints (that 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 [16]. Therefore, the following proposition holds.

Proposition 1: Problems (RMP^k) never generate the same integer solution twice.

Also, Problem (RMP^k) is a relaxation of the master problem (MP) when augmented with the relevant canonical integer cuts, in the sense that its feasible region is larger (due to the less constraints included), so it yields valid lower bound on the optimal objective value of the augmented master problem. In the case study in this paper, Problem (RMP^k) is further augmented into a tighter relaxation by constructing feasibility cuts for multiple scenarios respectively for each visited integer realizations (instead of constructing only one feasibility cut for that). The details are not shown here due to space limitations.

Notice that if $T^k = \emptyset$, Problem (RMP^k) is unbounded from below. In this case, any feasible solution of Problem (RMP^k) can be used to generate a new integer realization and let the algorithm proceed.

D. Decomposed Primal Subproblem

The solution of the lower bounding problem (LBP) yields a valid lower bound on the optimal objective value of the original problem (P). On the other hand, a valid upper bound for Problem (P) can be obtained by solving the primal problem with a particular integer realization, say $y = y^{(k)}$, which can naturally be decomposed into s subproblems. For each scenario h, the subproblem is as follows:

$$
\begin{aligned}
obj_{\mathrm{PP}_h}(y^{(k)}) = &\min_{x_h, q_h, u_h} c_{2,h}^{\mathrm{T}} x_h + c_{3,h}^{\mathrm{T}} q_h + c_{4,h}^{\mathrm{T}} u_h \\
\text{s.t.} \quad & u_{h,l,t} = x_{h,l} q_{h,t}, \quad \forall (l,t) \in \Omega, \\
& A_{1,h} y^{(k)} + A_{2,h} x_h + A_{3,h} q_h + A_{4,h} u_h \le b_h, \\
& (x_h, q_h, u_h) \in \Pi_h.
\end{aligned} \qquad (\mathrm{PP}_h)
$$

Problem (PP_h) is a nonconvex nonlinear program (NLP), which can be solved to ε-optimality in finite steps with a state-of-the-art deterministic global optimization method, such as branch-and-reduce [8]. In addition, the solution of Problem (PP_h) can be accelerated with the inclusion of an additional cut on the objective, which is derived from the solutions of the previous subproblems. (Again, although it is implemented for the case study, its details are not shown here due to space limitations). Nevertheless, Problem (PP_h) requires much more solution time than the decomposed primal bounding/feasible subproblems (which are only LPs), and it usually requires more solution time than the relaxed master problem (which is a MILP) as well. Therefore, the solution of Problem (PP_h) is postponed in the proposed decomposition strategy until the primal bounding problem solution and the relaxed master problem solution converge and a group of integer realizations have been examined for the lower bounding problem. Then, Problem (PP_h) is solved for these integer realizations to yield and update valid upper bounds on the target problem (P).

IV. ALGORITHM

A. Decomposition 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. Integer realization $y^{(1)}$ is given.

repeat

if $k = 0$ or (RMP^k is feasible and LBD $<$ UBDPB and LBD $<$ UBD) **then**

repeat

Set $k = k+1$

1. Solve the decomposed primal bounding problem $\mathrm{PBP}_h(y^{(k)})$ for each scenario $h = 1, \ldots, s$ sequentially. If $\mathrm{PBP}_h(y^{(k)})$ is feasible for all the scenarios with duality multipliers $\lambda_h^{(k)}$, add a cut to the relaxed master problem RMP^k according to $\lambda_1^{(k)}, \ldots, \lambda_s^{(k)}$, set $T^k = T^{k-1} \cup \{k\}$. If $obj_{\mathrm{PBP}}(y^{(k)}) = c_1^{\mathrm{T}} y^{(k)} + \sum_{h=1}^{s} obj_{\mathrm{PBP}_h}(y^{(k)}) <$ UBDPB, update UBDPB $= obj_{\mathrm{PBP}}(y^{(k)})$, $y^* = y^{(k)}$, $k^* = k$;
2. If $\mathrm{PBP}_h(y^{(k)})$ is infeasible for one scenario, stop solving it for the remaining scenarios and set $S^k = S^{k-1} \cup \{k\}$. Then, solve the decomposed feasibility problem $\mathrm{FP}_h(y^{(k)})$ for $h = 1, \ldots, s$ and obtain the corresponding Lagrange multiplier vector $\mu_h^{(k)}$. Add feasibility cuts to RMP^k according to these multipliers.
3. If $T^k = \emptyset$, solve the relaxed master problem RMP^k for a feasible solution; otherwise, solve RMP^k for an optimal solution. In the latter case, if RMP^k is feasible, set LBD to its optimal objective value and set $y^{(k+1)}$ to the y value at its optimum.

until LBD $\ge$ UBDPB or RMP^k is infeasible.

end if

if UBDPB $<$ UBD **then**

1. Solve the decomposed primal problem $\mathrm{PP}_h(y^*)$ for each scenario $h = 1, \ldots, s$ sequentially. Set $U^l = U^{l-1} \cup \{k^*\}$. If $\mathrm{PP}_h(y^*)$ is feasible with optimum $(x_{p,h}^*, q_{p,h}^*, u_{p,h}^*)$ for all the scenarios and $obj_{\mathrm{PP}}(y^*) = c_1^{\mathrm{T}} y^* + \sum_{h=1}^{s} obj_{\mathrm{PP}_h}(y^*) <$ UBD, update UBD $= obj_{\mathrm{PP}}(y^*)$ and $y_p^* = y^*$.
2. If $T^k \setminus U^l = \emptyset$, set UBDPB $= +\infty$.
3. If $T^k \setminus U^l \neq \emptyset$, pick the iteration index $i \in T^k \setminus U^l$ 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 $\ge$ UBD and (RMP^k is infeasible or LBD $\ge$ UBD).

The global optimum of the original problem P is given by $(y_p^*, x_{p,1}^*, \ldots, x_{p,s}^*, q_{p,1}^*, \ldots, q_{p,s}^*, u_{p,1}^*, \ldots, u_{p,s}^*)$ or P is infeasible.

B. Finite convergence

Definition 1: A feasible point of an optimization problem is an ε-optimal solution if it renders an objective value whose difference from the optimal objective value is within a particular tolerance ε. If an ε-optimal solution of the problem is obtained, the problem is said to be solved to ε-optimality.

Theorem 1: If all the subproblems (presented in Section III) can be solved to ε-optimality in a finite number of steps, the decomposition algorithm terminates in a finite number of steps providing an ε-optimal solution of Problem (P) or with an indication that Problem (P) is infeasible.

Proof: First, all the integer realizations are generated by solving Problem (RMP^k) in the algorithm, and according to Proposition 1, no integer realization will be generated twice. Since 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.

Next it is shown that, if Problem (P) is feasible, the algorithm terminates with an optimal solution of it. Notice that in this case the algorithm terminates with "Problem (RMP^k) is infeasible" or "LBD $\ge$ UBD". If Problem (RMP^k) is infeasible, then an optimum of Problem (P) is obtained at the integer realization $y = y_p^*$. (Note y_p^* exists because of the feasibility of Problem (P).) Then an optimal solution of the primal problem for $y = y_p^*$, which is constructed by the solutions of subproblems (PP_h) for all the scenarios, leads

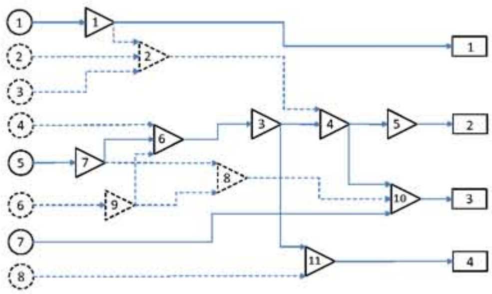

Fig. 1. Superstructure of the gas network.

to an optimal solution of Problem (P). If Problem (RMP^k) is feasible and LBD $\geq$ UBD, then a global optimum has been obtained by the solutions of the already visited Problem (RMP^k) and the corresponding subproblems (PP_h).

If Problem (P) is infeasible, the algorithm terminates with UBD $= +\infty$ because UBD can only be updated with optimal solutions of Problem (PP_h) for all the scenarios, which are not all feasible for any integer realization (and therefore UBD is never updated).

■

In practice, the convergence criteria, such as UBDPB $\geq$ UBD would not be used, but rather UBDPB $\geq$ UBD $- \varepsilon$, where ε is a predefined convergence tolerance. This is not explicitly addressed in the above algorithm and proof because of space limitation. However, the inclusion of the tolerance does not change the structure and the properties of the algorithm and the proof of finite convergence.

V. CASE STUDY

A. The Gas Network Planning Problem

The case study is to demonstrate the computational advantages of the proposed decomposition algorithm (DA) over BARON through a gas network planning problem. This problem is similar to the one studied in [2], which demonstrates the advantages of the stochastic pooling problem formulation over several deterministic formulations. Fig. 1 gives the superstructure of the gas network. The round symbols denote the sources of the gas flows which are typically gas wells, the triangular symbols denote the pools where the different gas flows are mixed and then split, which are typically gas production or riser platforms, the square symbols denote the product terminals where the final gas products are produced, which are typically plants supplying dry and/or liquefied gas products. The symbols with solid lines denote the facilities that must be developed for the network due to specific engineering reasons, and the symbols with dashed lines denote the potential facilities that can be developed for the system. The uncertainty in the network comes from the compositions of sources 2, 4 and 5, which are uniformly distributed within given ranges.

There are several differences between the problem studied here and the problem in [2]. First, source 7 and the pipeline connecting it to pool 10 must be developed while they are determined by the optimization in [2]. Second, component 1 content in each gas flow is not tracked and the bounds on it are not included in the problem. Third, additional integer constraints are added into the model which prevent any isolated node (i.e., a node that is not connected to any other node) to appear in the network. Finally, additional redudant constraints, which can tighten the relaxations of the bilinear terms (as suggested in [17]), are added into the model to accelerate the solution. Readers can find other details of the problem in [2].

The nonconvex MINLP problem to be solved for the gas network planning has 19 binary variables which decide the network design, and $68s$ (where s denotes the total number of scenarios addressed by the problem) continuous variables which determine the optimal operation for each scenario. So the size of the problem depends on the number of scenarios linearly.

The problem is solved on on a computer allocated a single 2.83GHz CPU and running Linux kernel. GAMS 22.8.1 is used to formulate the model, program the DA, and solve the problems with BARON and the DA. BARON 8.1.5 is used as the branch-and-reduce global optimization solver for comparison, which employs SNOPT 7.2.4 [18] for NLP subproblems and CPLEX 11.1.1 [19] for LP subproblems. The relative termination criteria for BARON is 1%. The DA employs BARON 8.1.5 with the same setting for NLP subproblems and CPLEX 11.1.1 for LP and MILP subproblems. The relative termination criteria for DA is 1% as well.

B. Results and Discussion

First, the uniformly distributed uncertain parameters in the model, i.e., the compositions of source 2, 4 and 5, are assumed to be correlated as described in [2]. Then, the scenarios in the stochastic formulation can be generated according to this assumption and the total number of scenarios addressed. When the number of scenarios is 1, the formulation is reduced to a deterministic formulation and the uncertain parameter is set to its expected value. Fig. 2 summarizes the total solver times with BARON and DA, respectively, for different numbers of scenarios. It is clear that, DA solves the problems faster than BARON does, and that its solution time increases moderately with the number of scenarios. On the other hand, the solution time with BARON increases exponentially with the number of scenarios. According to the trend shown in this figure, solving a problem with 100 scenarios would take BARON about 10^{60} CPU seconds! Fig. 2 also shows the solver times with explicit enumeration of integer realizations from set Y (called EI in the paper), which is estimated by the cardinality of Y and the solution time for Problem (PP_h). It can be seen that DA outperforms EI as well, because it only visits a small part of the elements of Y.

Second, it will be shown how DA performs for large numbers of scenarios. Here the three uniformly distributed uncertain parameters in the model are assumed to be independent. 2, 4, 6, 8, 10 scenarios are generated for each parameter respectively, which lead to problems with 8, 64, 216, 512, 1000 scenarios. Fig. 3 shows the solver timeswith

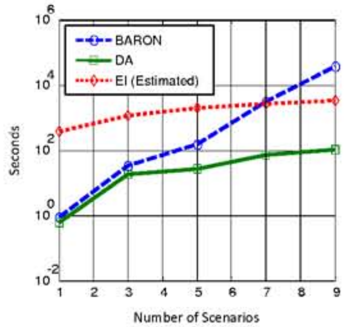

Fig. 2. Solver times for different numbers of scenarios.

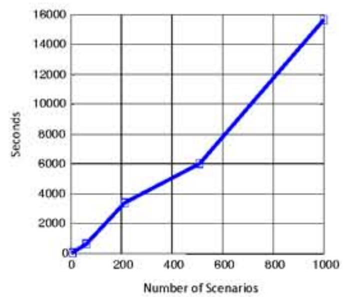

Fig. 3. Solver times with DA for more scenarios.

DA for these five cases. It can be seen that the increase in the solution time with DA is roughly linear with respect to the number of scenarios. Notice that the problem has 19 integer variables and 68000 continuous variables when addressing 1000 scenarios, which is a very large-scale MINLP. But this problem is solved by DA within only 5 hours solver time!

VI. CONCLUSIONS AND FUTURE WORK

This paper presents a rigorous decomposition method, that can guarantee an ε-optimal solution of the nonconvex MINLP problem for natural gas network design under uncertainty, with a finite number of iterations. Since this method can take advantage of the structure of the stochastic pooling problem, it has a tremendous advantage over state-of-the-art branch-and-bound based methods such as branch-and-reduce, and it also outperforms the explicit enumberation of integer realizations because it only visits part of the elements of the integer set. This is demonstrated by the case study of a gas network planning problem, where a nonconvex MINLP as large as with 68000 continuous variables and 19 binary variables is not solvable with BARON, but can be solved to ε-optimality by the decomposition method within only 5 hours solver time. The case study results also show that the increase of solution time with the proposed decomposition method is roughly linear with respect to the number of scenarios, which indicates the viability of the method for even larger problems.

Two interesting issues will be addressed in future work. First, notice that parallel computation naturally fits the decomposition algorithm because the decomposed primal subproblems, primal bounding subproblems and feasibility subproblems can be solved in parallel without exchanging any information between them. Integration of a parallel computing architecture can significantly reduce the run time of the method. Second, the proposed method can be extended to tackle more general nonconvex functions in the stochastic formulation, and thus it can solve more general nonconvex stochastic programming problems. Then, the decomposition method is not only applicable to integrated grid design including the natural gas subsystem, but also applicable to more types of grid design problems with some nonconvex functions in the model.

REFERENCES

[1] *Natural Gas Market Review 2007: Security in a globalizing market to 2015*. International Energy Agency, 2007.

[2] X. Li, E. Armagan, A. Tomasgard, and P. I. Barton, "Long-term planning of natural gas production systems via a stochastic pooling problem," in *Proceedings of the 2010 American Control Conference*, Baltimore, MD, USA, 2010.

[3] A. Quelhas, E. Gil, J. D. McCalley, and S. M. Ryan, "A multiperiod generalized network flow model of the U.S. integrated energy system: Part I - Model description," *IEEE Transactions on Power Systems*, vol. 22, pp. 829–836, 2007.

[4] *International Energy Outlook 2009*. U.S. Energy Information Administration, 2009.

[5] A. Quelhas and J. D. McCalley, "A multiperiod generalized network flow model of the U.S. integrated energy system: Part II - Simulation results," *IEEE Transactions on Power Systems*, vol. 22, pp. 829–836, 2007.

[6] S. An, Q. Li, and T. W. Gedra, "Natural gas and electricity optimal power flow," *Proceedings of the IEEE/PES Transmission and Distribution Conference 2003*, 2003.

[7] H. Hashim, P. Douglas, A. Elkamel, and E. Croiset, "Optimization model for energy planning with CO_2 emission considerations," *Industrial and Engineering Chemistry Research*, vol. 44, pp. 879–890, 2005.

[8] H. S. Ryoo and N. V. Sahinidis, "A branch-and-reduce approach to global optimization," *Journal of Global Optimization*, vol. 8, pp. 107–138, 1996.

[9] C. S. Adjiman, I. P. Androulakis, and C. A. Floudas, "Global optimization of mixed-integer nonlinear problems," *AIChE Journal*, vol. 46, no. 9, pp. 1769–1797, 2000.

[10] M. Duran and I. E. Grossmann, "An outer-approximation algorithm for a class of mixed nonlinear programs," *Mathematical Programming*, vol. 66, pp. 327–349, 1986.

[11] R. Fletcher and S. Leyffer, "Solving mixed integer nonlinear programs by outer approximation," *Mathematical Programming*, vol. 66, pp. 327–349, 1994.

[12] 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*, vol. 100, pp. 517–535, 2004.

[13] A. M. Geoffrion, "Generalized Benders decomposition," *Journal of Optimization Theory and Applications*, vol. 10, no. 4, pp. 237–260, 1972.

[14] ——, "Elements of large-scale mathematical programming: Part I: Concepts." *Management Science*, vol. 16, no. 11, pp. 652–675, 1970.

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

[16] E. Balas and R. Jeroslow, "Canonical cuts on the unit hypercube," *SIAM Journal on Applied Mathematics*, vol. 23, no. 1, pp. 61–69, 1972.

[17] M. Tawarmalani and N. Sahinidis, *Convexification and global optimization in continuous and mixed-integer nonlinear programming*. Dordrecht, the Netherlands: Kluwer Academic Publishers, 2002.

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

[19] IBM, "IBM ILOG CPLEX: High-performance mathematical programming engine," http://www-01.ibm.com/software/integration/optimization/cplex/.