A new Decomposition Algorithm for Multistage Stochastic Programs with Endogenous Uncertainties

Vijay Gupta*, Ignacio E. Grossmann†,
Department of Chemical Engineering, Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

In this paper, we present a new decomposition algorithm for solving large-scale multistage stochastic programs (MSSP) with endogenous uncertainties. Instead of dualizing all the initial non-anticipativity constraints (NACs) and removing all the conditional non-anticipativity constraints to decompose the problem into scenario subproblems, the basic idea relies on keeping a subset of NACs as explicit constraints in the scenario group subproblems while dualizing or relaxing the rest of the NACs. It is proved that the algorithm provides a dual bound that is at least as tight as the standard approach. Numerical results for process network examples and oilfield development planning problem are presented to illustrate that the proposed decomposition approach yields significant improvement in the dual bound at the root node and reduction in the total computational expense for closing the gap.

Keywords: multistage stochastic programming; endogenous uncertainties; non-anticipativity constraints; Lagrangean decomposition; process networks; oil & gas exploration

1. Introduction

Stochastic programming is typically used to model problems where some of the parameters are random (e.g. uncertain reservoir size, product demand, yields, prices), Birge and Louveaux (1997). In general, multiperiod industrial planning, scheduling, supply-chain etc. problems under uncertainty are formulated as stochastic programs since it allows to incorporate probability distribution of the uncertain parameters explicitly into the model while making investment and

*E-mail: vijaygup@andrew.cmu.edu
†To whom all correspondence should be addressed. E-mail: grossmann@cmu.edu

operations decisions, and provides an opportunity to take corrective actions in the future (recourse) based on the actual outcomes (see Ierapetritou and Pistikopoulos, 1994; Clay and Grossmann, 1997; Iyer and Grossmann, 1998; Schultz, 2003; Ahmed and Garcia, 2003; Sahinidis, 2004; Ahmed et al. 2004; Li and Ierapetritou, 2012). Discrete probability distributions of the uncertain parameters that give rise to scenarios are widely considered to represent scenarios that are given by combinations of the realization of the uncertain parameters. Depending on the number of decision stages involved in the model, the stochastic program corresponds to either a two-stage or a multistage problem. The main idea behind two-stage stochastic programming is that we make some decisions (stage 1) here and now based on not knowing the future outcomes of the uncertain parameters, while the rest of the decisions are stage -2 (recourse actions) decisions are made after uncertainty in those parameters is revealed. In this paper, we focus on more general multistage stochastic programming models where the uncertain parameters are revealed sequentially, i.e. in multiple stages (time periods), and the decision-maker can take corrective actions over a sequence of the stages. In the two-stage and multistage case the cost of the decisions and the expected cost of the recourse actions are optimized.

Based on the type of uncertain parameters involved in the problem, stochastic programming models can be classified into two broad categories (Jonsbraten, 1998): exogenous uncertainty where stochastic processes are independent of decisions that are taken (e.g. demands, prices), and endogenous uncertainty where stochastic processes are affected by these decisions (e.g. reservoir size and its quality). Our decisions can affect the stochastic processes in two different ways (Goel and Grossmann, 2006): either they can alter the probability distributions (type 1) (see Viswanath et al., 2004; and Held and Woodruff, 2005), or they can determine the timing when uncertainties in the parameters are resolved (type 2) (see Goel et al., 2006; Gupta and Grossmann, 2011). A number of planning problems involving very large investments at an early stage of the project have endogenous (technical) uncertainty (type 2) that dominates the exogenous (market) uncertainty. In such cases, it is essential to incorporate endogenous uncertain parameters while making the investment decisions since it can have a large impact on the overall project profitability. Surprisingly, these problems have received relatively little attention in the literature despite their practical importance.

In this paper, we focus on the type 2 of endogenous uncertainty where the decisions are used to gain more information, and resolve uncertainty either immediately or in a gradual manner. Therefore, the resulting scenario tree is decision-dependent that requires modeling a superstructure of all possible scenario trees that can occur based on the timing of the decisions. In this context, we present a multistage stochastic programming framework to model the problems in this class in which special disjunctive constraints with propositional logic are considered to enforce the conditional non-anticipativity constraints that define the decision-dependent scenario tree. Recently, few practical applications that involve multistage stochastic programming with endogenous uncertainty have been addressed: Goel and Grossmann (2004), and Goel et al. (2006) for gas field development planning; Tarhan et al. (2009), and Gupta and Grossmann (2013) for oil/gas field investments and operations; Tarhan and Grossmann (2008), and Gupta and Grossmann (2011) for process networks planning; Solak (2007) for project portfolio optimization; Boland et al. (2008) for open pit mine scheduling; and Colvin and Maravelias (2008) for pharmaceutical testing.

In general, these multistage stochastic programs become very difficult to solve directly as deterministic equivalent since the problem size (constraints and variables) increases with the number of scenarios, whereas the solution time increases exponentially. Therefore, special solution techniques are used to solve problems in this class. Several fullspace based approaches for the medium-size problems exploiting the properties of the model and the optimal solution have been proposed. In particular, Colvin and Maravelias (2010) developed a branch and cut framework, while Gupta and Grossmann (2011) proposed a NAC relaxation strategy to solve these MSSP problems under the assumption that only few non-anticipativity constraints be active at the optimal solution.

Lagrangean decomposition is a widely used technique to solve large-scale problems that have decomposable structure as in stochastic programs (Fisher, 1985; Ruszczynski, 1997; Caroe and Schultz, 1999; Guignard, 2003; Conejo et al. 2006). It addresses problems where a set of constraints links several smaller subproblems. If these constraints are removed by dualizing them, the resulting subproblems can be solved independently. In the case of multistage stochastic programs with endogenous uncertainty initial and conditional non-anticipativity constraints are the linking constraints, while each subproblem corresponds to the problem for a given scenario. Therefore, the model has the decomposable structure that is amenable to Lagrangean decomposition approaches. In this context, a Lagrangean decomposition algorithm based on dualizing all the initial NACs and relaxing all the conditional NACs that allow parallel solution of the scenario subproblems has been proposed by Gupta and Grossmann (2011). An extended form of this decomposition approach relying on the duality based branch and bound search is also presented in Goel and Grossmann (2006), Tarhan et al. (2009), and Tarhan et al. (2011) to close the gap between the upper and lower bounds. Solak (2007) used a sample average approximation method for solving the problem in this class, where the sample problems were solved through Lagrangean relaxation and heuristics. However, there are several limitations with these methods including a weak dual bound at the root node, a large number of iterations to converge at each node, and many nodes that may be required during the branch and bound search to close the gap depending on the branching rules and variables. Moreover, the number of subproblems to be solved during each iteration at every node grows linearly with the number of scenarios. In this work, we propose a new decomposition scheme for solving these multistage stochastic programs that overcomes some of the limitations of the standard approaches.

The outline of this paper is as follows. First, we introduce the problem statement with particular focus on the problems where timing of uncertainty realization depends on the optimization decisions. Then, a generic mixed-integer linear multistage stochastic disjunctive programming model for endogenous uncertainty problems is presented. Several Lagrangean decomposition approaches that have been used and their limitations are identified next. To overcome these limitations, we propose a new Lagrangean decomposition scheme that relies on the concept of scenario group partitions. Numerical results of process networks and oilfield planning problems are presented for the various decomposition approaches.

2. Problem Statement

We focus here on multiperiod planning problems that have endogenous uncertainty in some the parameters, i.e. where timing of uncertainty realization depends on our decisions. In particular, the time horizon is represented by the discrete set of time periods $T = \{1, 2, . . . , T\}$. The set of endogenous uncertain parameters $\Theta = \{\theta_1, \theta_2, ...\}$ is considered where each parameter has a discrete set of possible realizations. Therefore, a scenario s represents the possible combination of the realizations of these uncertain parameters with a probability p^s. Note that when some of the parameters θ_p are correlated as they may belong to a particular uncertainty source, then the resulting scenario set will be smaller. The timing of uncertainty resolution in each uncertain parameter depends on the decisions x_t^s (both discrete and continuous) that have been implemented so far. Furthermore, the uncertainty resolution rule can be immediate (Goel and Grossmann, 2006; and Gupta and Grossmann, 2011) or gradual (Tarhan et al., 2009) depending on the problem at hand. Therefore, the resulting scenario tree is decision-dependent, and hence we need to use a superstructure of all possible scenario-trees that can occur based on the decisions. In particular, we use logic propositions and disjunctions as in Goel and Grossmann (2006) and Gupta and Grossmann (2011) to represent the scenario-tree for the problems in this class. The uncertainty realizations for each parameter θ_p are assumed to be time invariant. In the next section, we present a MSSP model corresponding to this description.

3. Model

A mixed-integer linear disjunctive multistage stochastic program with endogenous uncertainties can be represented in the following compact form:

(MD)

$$\min \quad z = \sum_{s \in S} p^s \sum_{t \in T} c_t x_t^s \tag{1}$$

$$s.t. \quad \sum_{\tau \le t} A_\tau^s x_\tau^s \le a_t^s \qquad \forall t, s \tag{2}$$

$$x_t^s = x_t^{s'} \quad \forall t \in T_I, \forall s, s' \in S \tag{3}$$

$$Z_t^{s,s'} \Leftrightarrow F(x_1^s, x_2^s \ldots x_{t-1}^s) \quad \forall t \in T_C, \forall s, s' \in S \tag{4}$$

$$\begin{bmatrix} Z_t^{s,s'} \\ x_t^s = x_t^{s'} \end{bmatrix} \vee \begin{bmatrix} \neg Z_t^{s,s'} \end{bmatrix} \qquad \forall t \in T_C, \forall s, s' \in S \tag{5}$$

$$x_{jt}^s \in \{0,1\} \qquad \forall t, s, \forall j \in J' \tag{6}$$

$$x_{jt}^s \in R \qquad \forall t, s, \forall j \in J \setminus J' \tag{7}$$

The objective function (1) in the above model **(MD)** minimizes the expectation of an economic criterion over the set of scenarios $s \in S$, and over a set of time periods $t \in T$. For a particular scenario *s*, inequality (2) represents constraints that govern decisions x_t^s in time period *t* and link decisions across time periods. Non-anticipativity (NA) constraints for initial time periods $T_I \subset T$ are given by equations (3) for each scenario pair *(s,s')* to ensure the same decisions in all the scenarios. The conditional NA constraints are written for the later time periods $T_C \subset T$ in terms of logic propositions (4) and disjunctions (5). Notice that the set of initial time periods T_I may include first few years of the planning horizon until uncertainty cannot be revealed, while T_C represents the rest of the time periods in the planning horizon. The function $F(x_1^s, x_2^s \ldots x_{t-1}^s)$ in eq. (4) is an uncertainty resolution rule for a given pair of scenarios *s* and *s'* that determines the value of the corresponding boolean variable $Z_t^{s,s'}$ based on the decisions that have been implemented so far. The variable $Z_t^{s,s'}$ is further used in disjunction (5) to ensure the same decisions in scenarios *s* and *s'* if these are still indistinguishable in time period *t*. Eqs. (6)-(7) define the domain of the discrete and continuous variables in the model.

Notice that the model with reduced number of scenario pairs *(s,s')* that are sufficient to represent the non-anticipativity constraints can be obtained from model (MD) after applying the three properties presented in the paper by Gupta and Grossmann (2011). These properties are defined on the basis of symmetry, adjacency and transitivity relationship among the scenarios. The reduced model **(MDR)** can be formulated as follows, where P_3 is the set of minimum number of scenario pairs that are required to represent non-anticipativity in each time period *t*,

(MDR)

$$\min \quad z = \sum_{s \in S} p^s \sum_{t \in T} c_t x_t^s \tag{1}$$

$$s.t. \quad \sum_{\tau \le t} A_\tau^s x_\tau^s \le a_t^s \qquad \forall t, s \tag{2}$$

$$x_t^s = x_t^{s'} \quad \forall t \in T_I, \forall (s,s') \in P_3 \tag{3a}$$

$$Z_t^{s,s'} \Leftrightarrow F(x_1^s, x_2^s \ldots x_{t-1}^s) \quad \forall t \in T_C, \forall (s,s') \in P_3 \tag{4a}$$

$$\begin{bmatrix} Z_t^{s,s'} \\ x_t^s = x_t^{s'} \end{bmatrix} \vee \begin{bmatrix} \neg Z_t^{s,s'} \\ \end{bmatrix} \qquad \forall t \in T_C, \forall (s,s') \in P_3 \tag{5a}$$

$$x_{jt}^s \in \{0,1\} \qquad \forall t, s, \forall j \in J' \tag{6}$$

$$x_{jt}^s \in R \qquad \forall t, s, \forall j \in J \setminus J' \tag{7}$$

We then define the following sets,

$$L_p = \left\{ (s_1, s_2, \ldots, s_k) \middle| s_1, s_2, \ldots, s_k \in S, s_1 < s_2 < \ldots < s_k, D(s,s') = \{p\} \ \forall (s,s') \in (s_1, s_2, \ldots, s_k) \right\} \quad \forall \theta_p \in \Theta \tag{8}$$

$$D(s,s') = \left\{ p \middle| \theta_p \in \Theta, \hat{\theta}_p^s \neq \hat{\theta}_p^{s'} \right\} \tag{9}$$

$$P_3 = \left\{ (s_1,s_2),(s_2,s_3),\ldots,(s_{k-1},s_k) \middle| (s_1, s_2, \ldots, s_k) \in L_p \ \forall \theta_p \in \Theta \right\} \tag{10}$$

Notice that the minimum scenario pair set $(s,s') \in P_3$ can be obtained by first defining a scenario group set $(s_1, s_2, \ldots, s_k) \in L_p$ for each uncertain parameter $\theta_p \in \Theta$ with k realizations (eq. 8) such that the k scenarios in each of these $(s_1, s_2, \ldots, s_k)$ set can only be realized at the same time irrespective of the other realizations during the given time horizon. The basic idea to identify such scenario sets $(s_1, s_2, \ldots, s_k)$ is that all the scenarios in each of these sets only differ in the realization of the uncertain parameter θ_p for which the corresponding set is defined. Therefore, for any scenario pair $(s,s') \in (s_1, s_2, \ldots, s_k)$, the value of $D(s,s') = \{p\}$ where $D(s,s')$ represents the index of the uncertain parameter $\theta_p \in \Theta$ in eq. (9) that distinguish the two scenarios *s* and *s'* having values $\hat{\theta}_p^s$ and $\hat{\theta}_p^{s'}$, respectively. The required minimum scenario pair set P_3 (eq. 10) then corresponds to the consecutive elements in the scenario group sets $(s_1, s_2, \ldots, s_k) \in L_p$ for each uncertain parameter $\theta_p \in \Theta$. The cardinality of the set P_3 is $|\Theta|(|S| - |S|^{|\Theta|-1/|\Theta|})$ as shown in Gupta and Grossmann (2011).

For instance, if there are 2 uncertain parameters, *i.e.* (θ_1, θ_2). Each of these uncertain parameters has three realizations (L, M, H) which give rise to a total of 9 scenarios shown in Table 1. The original model (MD) requires a total of 72 scenario pairs (Table 2a) to represent the non-anticipativity, while the reduced model (MDR) only requires 12 scenario pairs (Table 2b), i.e. $|P_3| = 12$ in each time period *t* (see Gupta and Grossmann (2011) for details).

Table 1: 2 uncertain parameters, 9 scenarios

Scenario (s)	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
θ_1	L	M	H	L	M	H	L	M	H
θ_2	L	L	L	M	M	M	H	H	H

Table 2: Scenario pairs for the original and the reduced model

D(s,s')	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
1		1	1	2	1,2	1,2	2	1,2	1,2
2	1		1	1,2	2	1,2	1,2	2	1,2
3	1	1		1,2	1,2	2	1,2	1,2	2
4	2	1,2	1,2		1	1	2	1,2	1,2
5	1,2	2	1,2	1		1	1,2	2	1,2
6	1,2	1,2	2	1	1		1,2	1,2	2
7	2	1,2	1,2	2	1,2	1,2		1	1
8	1,2	2	1,2	1,2	2	1,2	1		1
9	1,2	1,2	2	1,2	1,2	2	1	1	

(a) 72 Scenario pairs in the original model (MD)

D(s,s')	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
1		1		2					
2			1		2				
3						2			
4					1		2		
5						1		2	
6									2
7								1	
8									1
9									

(b) 12 Scenario pairs in the reduced model (MDR)

The mixed-integer linear disjunctive model (MDR) can further be converted to a mixed-integer linear programming model **(MLR)**. First, the logic constraints (4a) are re-written as the mixed-integer linear constraints eq. (4b) based on the uncertainty resolution rule, where $z_t^{s,s'}$ is a binary variable that takes a value of 1 if scenario pair *(s,s')* is indistinguishable in time period *t*, and zero otherwise. The disjunction (5a) can then be converted to mixed-integer linear constraints (5b) and (5c) using the big-M formulation. The resulting mixed-integer linear model (MLR) includes constraints (1), (2), (3a), (4b), (5b), (5c), (6) and (7).

$$B_t^s x_t^s + C_t^s z_t^{s,s'} \le d_t^s \qquad \forall t \in T_C, \forall (s,s') \in P_3 \tag{4b}$$

$$-M(1 - z_t^{s,s'}) \le x_t^s - x_t^{s'} \quad \forall t \in T_C, \forall (s,s') \in P_3 \tag{5b}$$

$$M(1 - z_t^{s,s'}) \ge x_t^s - x_t^{s'} \quad \forall t \in T_C, \forall (s,s') \in P_3 \tag{5c}$$

Figure 1 represents the block angular structure of model (MLR), where we can observe that the initial (eq. (3a)) and conditional (eqs. (4b), (5b) and (5c)) non-anticipativity constraints link the scenario subproblems (eq. (2)), i.e. these are the complicating constraints in the model. However, this structure allows decomposing the fullspace problem into smaller subproblems by relaxing the linking constraints. It should be noted that the NACs (especially conditional NACs) represent a large fraction of the total constraints in the model.

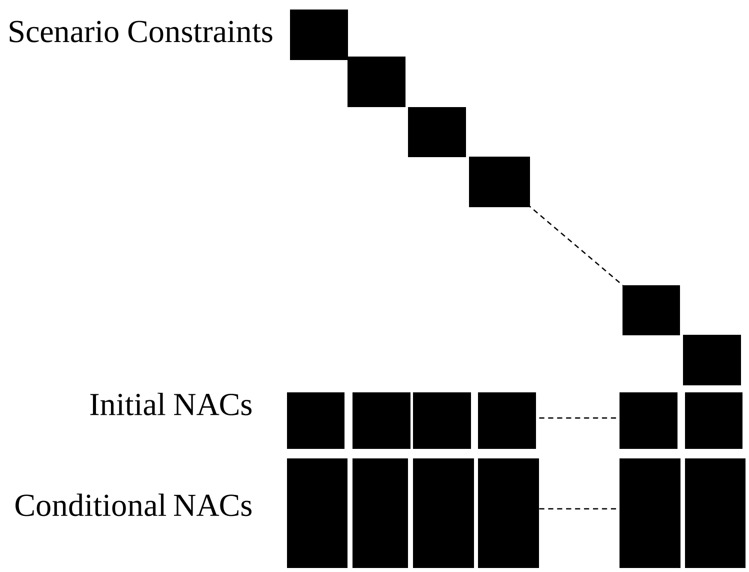

Figure 1: Structure of a typical Multistage Stochastic Program with Endogenous uncertainties

4. Conventional Lagrangean Decomposition Algorithms

The reduced model (MLR) is composed of scenario subproblems connected through initial (eq. (3a)) and conditional (eq. (4b), (5b) and (5c)) NA constraints. If these NA constraints are either relaxed or dualized using Lagrangean decomposition, then the problem decomposes into smaller subproblems that can be solved independently for each scenario within an iterative scheme for the multipliers as described in Caroe and Schultz (1999) and in Gupta and Grossmann (2011). In this way, we can effectively decompose the large scale problems in this class. However, there are several decomposition schemes that can be used for this structure (Figure 1) as described below:

4.1.1 Lagrangean Decomposition based on relaxing conditional NACs (Standard approach): In the decomposition algorithm of Figure 2 for MSSP with endogenous uncertainties as proposed in Gupta and Grossmann (2011), the lower bound (LB) is obtained by solving the Lagrangean problem with fixed multipliers $\lambda_t^{s,s'}$,

(L1-MLR)

$$\min \sum_{s\in S} p^s \sum_{t\in T} c_t x_t^s + \sum_{t\in T_1} \sum_{(s,s')\in P_3} \lambda_t^{s,s'}(x_t^s - x_t^{s'}) \quad (1a)$$

$$s.t. \quad \sum_{\tau\le t} A_\tau^s x_\tau^s \le a_t^s \qquad \forall t, s \quad (2)$$

$$x_{jt}^s \in \{0,1\} \qquad \forall t, s, \forall j \in J' \quad (6)$$

$$x_{jt}^s \in R \qquad \forall t, s, \forall j \in J \setminus J' \quad (7)$$

which gives rise to the subproblems for each scenario $s \in S$,

(L1-MLR^s)

$$\min \sum_{t\in T} p^s c_t x_t^s + \sum_{t\in T_1} x_t^s \Big(\sum_{\substack{(s,s')\in P_3 \\ s<s'}} \lambda_t^{s,s'} - \sum_{\substack{(s',s)\in P_3 \\ s>s'}} \lambda_t^{s',s} \Big) \quad (1b)$$

$$s.t. \quad \sum_{\tau\le t} A_\tau^s x_\tau^s \le a_t^s \qquad \forall t \quad (2a)$$

$$x_{jt}^s \in \{0,1\} \qquad \forall t, \forall j \in J' \quad (6a)$$

$$x_{jt}^s \in R \qquad \forall t, \forall j \in J \setminus J' \quad (7a)$$

In particular, the Lagrangean problem **(L1-MLR)** is formulated from the mixed-integer linear reduced model (MLR) by relaxing all the conditional NA constraints (4b), (5b) and (5c) and dualizing all the initial NA constraints (3a) as penalty terms in the objective function. Figure 3 represents the structure of the resulting model (L1-MLR). Notice that the each sub-problem **(L1-MLR^s)** in the Lagrangean problem (L1-MLR) corresponds to a scenario that can be solved in parallel.

The upper bound (UB) is generated by using a heuristic based on the solution of the Lagrangean problem (L1-MLR). In this heuristic, we fix the decisions obtained from the above problem (L1-MLR) in the reduced problem (MLR) such that there is no violation of NA constraints and solve it to obtain the upper bound. The sub-gradient method by Fisher (1985) or an alternative update scheme (see Mouret et al., 2011; Oliveira et al., 2013; and Tarhan et al. 2013) is used during each iteration to update the Lagrangean multipliers. The algorithm stops when either a maximum iteration/time limit is reached, or the difference between the lower and upper bounds, LB and UB, is less than a pre-specified tolerance. Notice that the extended form of this method relying on duality based branch and bound search has also been proposed in Goel and Grossmann (2006); Tarhan et al. (2009), and Tarhan et al. (2011) to close the gap between the upper and the lower bounds.

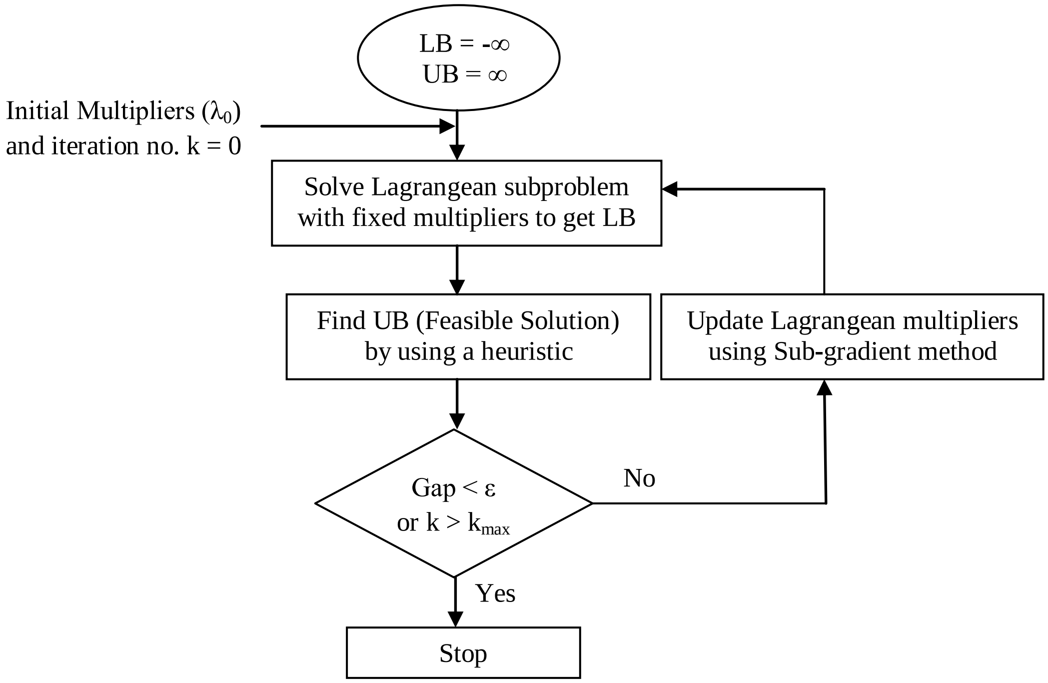

Figure 2: Lagrangean Decomposition algorithm

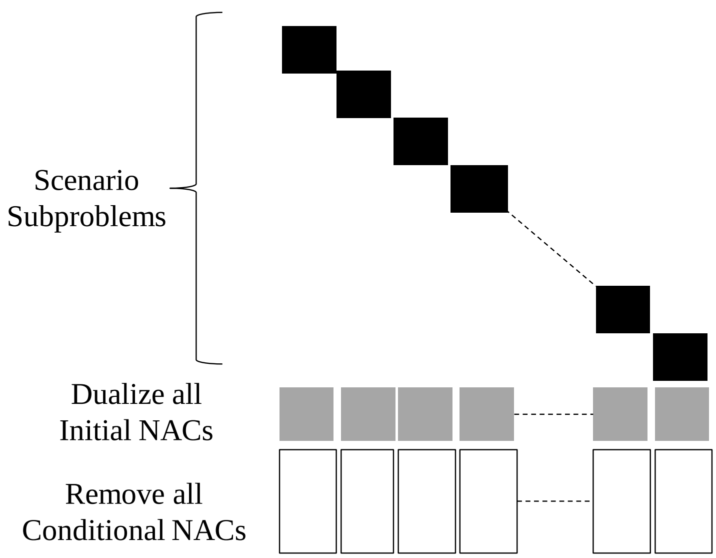

Figure 3: Lagrangean Decomposition based on relaxing conditional NACs

4.1.2 Limitations: We can observe from Figure 3 that the major limitation of this Lagrangean Decomposition algorithm for endogenous uncertainty problems (Gupta and Grossmann, 2011; Goel and Grossmann, 2006; Tarhan et al., 2009; and Tarhan et al., 2011) is that all the conditional non-anticipativity constraints (4b), (5b) and (5c) are removed while formulating the scenario subproblems at the root node. These constraints represent a large fraction of the total constraints in the model and can have significant impact on the decisions. For instance, in Figure 4, the scenario tree for the later time periods T_C (conditional NACs) can be constructed in several ways even though the initial NACs (for time periods T_I) are satisfied.

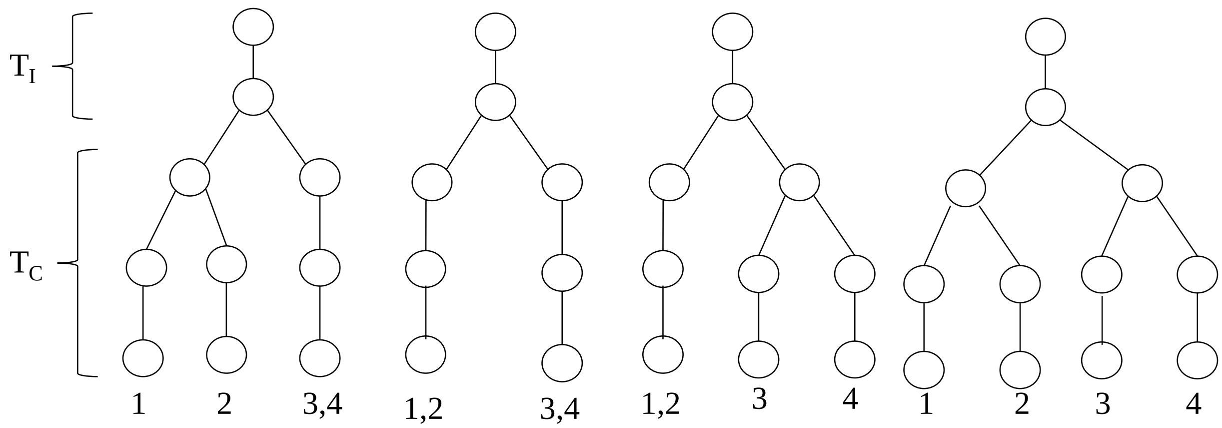

Figure 4: Impact of conditional NACs on the scenario tree structure

Therefore, there can be several undesired consequences that can occur with this relaxation approach:

1. The dual bound at root node can be significantly weaker since a large amount of information from the conditional NACs is ignored. In particular, only the initial NAC are considered (dualized) while formulating the subproblems at the root node, which represent only a first few time periods in the model. This means that the dynamics of the problem corresponding to the later periods is completely relaxed.
2. It is theoretically impossible to obtain a dual bound that is stronger than the optimal solution of the model without all conditional NACs at the root node.
3. The total number of nodes in the branch and bound search tree and the number of iterations required at each node can be very large.
4. Since many constraints are relaxed form the model, a good heuristic is needed to generate a feasible solution based on the solution of the dual problem.
5. The number of subproblems grows with the number of uncertain parameters and their realizations in an exponential manner.
6. It is problem specific and non-intuitive to define the branching rules/variables in the tree search since there are several alternatives.

4.2.1 Lagrangean Decomposition based on Dualizing all the NACs:

(i) In this decomposition approach, we dualize all the NACs (both initial (3a) and conditional (5b) and (5c)) in the objective function directly while formulating the lower bounding Lagrangean problem **(L2-MLR)**, which is still decomposable into individual scenarios. Notice that since (5b) and (5c) are inequality constraints, the corresponding Lagrangean multipliers $\lambda_{tg}^{s,s'}$ and $\lambda_{tl}^{s,s'}$ need to be non-negative.

(L2-MLR)

$$\min \sum_{s\in S} p^s \sum_{t\in T} c_t x_t^s + \sum_{t\in T_1} \sum_{(s,s')\in P_3} \lambda_t^{s,s'}(x_t^s - x_t^{s'})$$
$$+\sum_{t\in T_C} \sum_{(s,s')\in P_3} \lambda_{tg}^{s,s'}(x_t^{s'} - x_t^s - M(1-z_t^{s,s'}))$$
$$+\sum_{t\in T_C} \sum_{(s,s')\in P_3} \lambda_{tl}^{s,s'}(x_t^s - x_t^{s'} - M(1-z_t^{s,s'})) \qquad (1c)$$

$$s.t. \quad \sum_{\tau\le t} A_\tau^s x_\tau^s \le a_t^s \qquad \forall t,s \qquad (2)$$

$$B_t^s x_t^s + C_t^s z_t^{s,s'} \le d_t^s \qquad \forall t\in T_C, \forall (s,s')\in P_3 \qquad (4b)$$

$$x_{jt}^s \in \{0,1\} \qquad \forall t,s,\forall j\in J' \qquad (6)$$

$$x_{jt}^s \in R \qquad \forall t,s,\forall j\in J\setminus J' \qquad (7)$$

Figure 5 represents the structure of the model (L2-MLR) where **L2-MLR^s** correspond to the scenario sub-problems in this decomposed model.

(L2-MLR^s)

$$\min \sum_{t\in T} p^s c_t x_t^s + \sum_{t\in T_1} x_t^s \Big(\sum_{\substack{(s,s')\in P_3\\ s<s'}} \lambda_t^{s,s'} - \sum_{\substack{(s',s)\in P_3\\ s>s'}} \lambda_t^{s',s} \Big)$$
$$+\sum_{t\in T_C} x_t^s \Big[\sum_{\substack{(s,s')\in P_3\\ s<s'}} (\lambda_{tl}^{s,s'} - \lambda_{tg}^{s,s'}) - \sum_{\substack{(s',s)\in P_3\\ s>s'}} (\lambda_{tl}^{s',s} - \lambda_{tg}^{s',s}) \Big] \qquad (1d)$$
$$-\sum_{t\in T_C} \sum_{\substack{(s,s')\in P_3\\ s<s'}} [(1-z_t^{s,s'}).(\lambda_{tg}^{s,s'} + \lambda_{tl}^{s,s'}).M)$$

$$s.t. \quad \sum_{\tau \le t} A_\tau^s x_\tau^s \le a_t^s \qquad \forall t \tag{2a}$$

$$B_t^s x_t^s + C_t^s z_t^{s,s'} \le d_t^s \qquad \forall t \in T_C, \forall (s,s') \in P_3, s < s' \tag{4c}$$

$$x_{jt}^s \in \{0,1\} \qquad \forall t, \forall j \in J' \tag{6a}$$

$$x_{jt}^s \in R \qquad \forall t, \forall j \in J \setminus J' \tag{7a}$$

It is important to observe that we assign the shared binary variable $z_t^{s,s'}$ and its corresponding constraints (4b) and objective function term to the scenario problem *s* for all $(s,s') \in P_3$ where ($s < s'$). This allows to decompose the problem into independent scenarios. For instance in the case of 4 scenarios, the minimum scenario pair set $P_3 = \{(1,2),(1,3),(2,4),(3,4)\}$ and, therefore, the corresponding shared variables $z_t^{1,2}, z_t^{1,3}$ are assigned to scenarios 1; $z_t^{2,4}$ to scenario 2; and $z_t^{3,4}$ to scenario 3. As an alternative, one can also create a copy of the shared variable $z_t^{s,s'}$ as $z_t^{s',s}$ and its corresponding constraints (4b), (5b) and (5c) for all $(s,s') \in P_3$, that will allow to keep these variables in both the sub-problems *s* and *s'*. However, the performance of the two alternative decomposition approaches should not be very different.

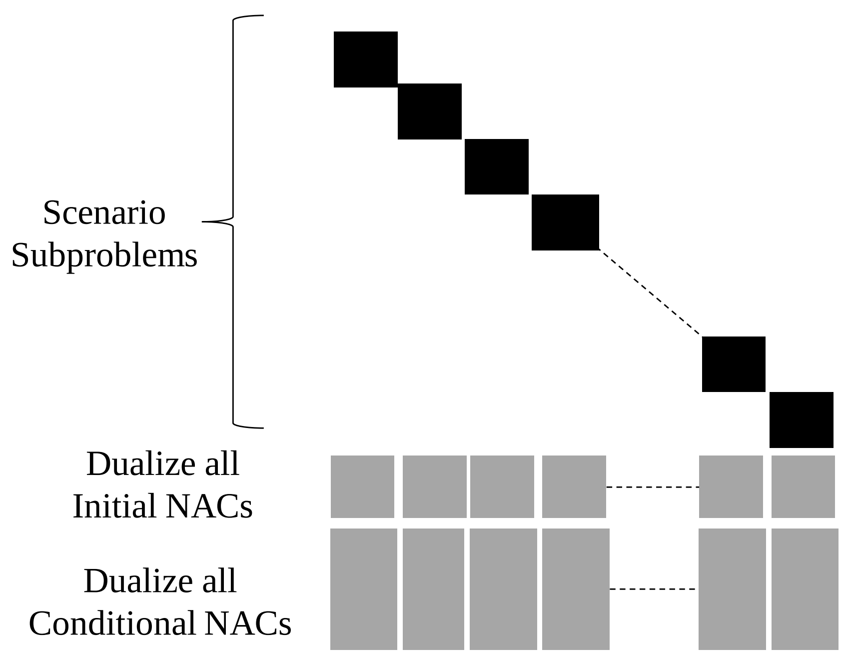

Figure 5: Lagrangean Decomposition based on dualizing all NACs directly

(ii) Another way to decompose the model (MLR) while considering all the NACs, is based on first reformulating the constraints (3a), (5b) and (5c) as (3b), (5d) and (5e) respectively, where $\tilde{x}_t^{s,s'}$ represents the value of the variable $x_t^{s'}$ for $\forall t \in T, \forall (s,s') \in P_3$.

$$x_t^s = \tilde{x}_t^{s,s'} \quad \forall t \in T_I, \forall (s,s') \in P_3 \tag{3b}$$

$$-M(1 - z_t^{s,s'}) \quad \leq x_t^s - \tilde{x}_t^{s,s'} \quad \forall t \in T_C, \forall (s,s') \in P_3 \tag{5d}$$

$$M(1 - z_t^{s,s'}) \quad \geq x_t^s - \tilde{x}_t^{s,s'} \quad \forall t \in T_C, \forall (s,s') \in P_3 \tag{5e}$$

$$x_t^{s'} = \tilde{x}_t^{s,s'} \quad \forall t \in T, \forall (s,s') \in P_3, s < s' \tag{5f}$$

In addition, eq. (5f) is required to ensure that all the copy variables $\tilde{x}_t^{s,s'}$ for $x_t^{s'}$ have the same values in all the scenario pairs it occurs. Notice that the reformulated model (MLR^C) includes constraints (1), (2), (3b), (4b), (5d), (5e), (5f), (6) and (7). Model (MLR^C) can now be decomposed into individual scenarios by dualizing only constraints (5f) as can be seen in Figure 6. **L3-MLR^C** and **L3-MLR^{Cs}** represent the Lagrangean problem and scenario sub-problems for this indirect decomposition approach, respectively.

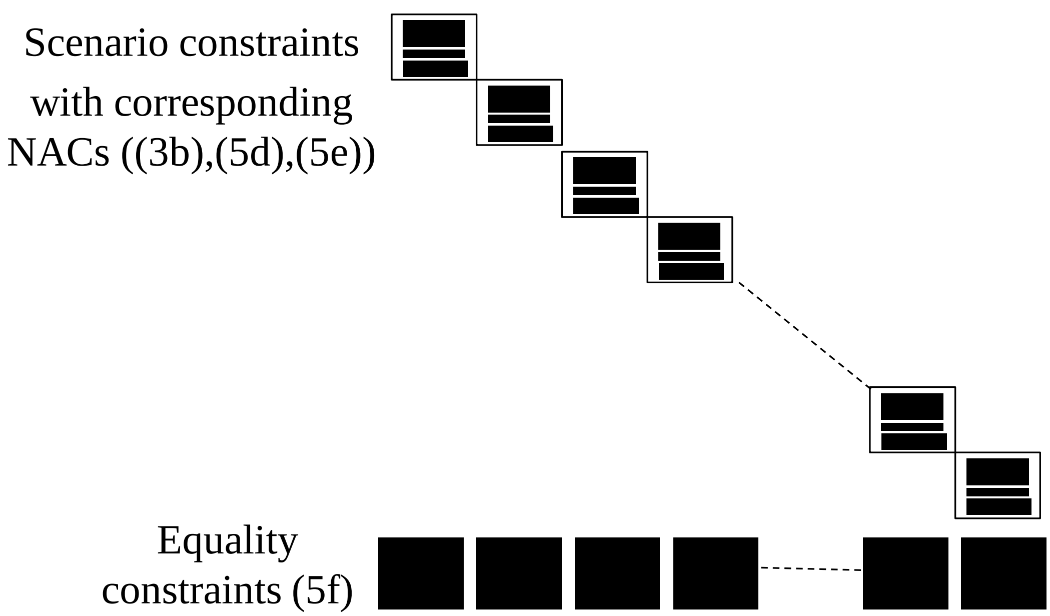

Figure 6: Structure of the Reduced Model after reformulation (MLR^C)

(L3-MLR^C)

$$\min \quad \sum_{s \in S} p^s \sum_{t \in T} c_t x_t^s + \sum_{t \in T} \sum_{\substack{(s,s') \in P_3 \\ s<s'}} \lambda_t^{s,s'} (x_t^{s'} - \tilde{x}_t^{s,s'}) \tag{1e}$$

$$s.t. \quad \sum_{\tau \le t} A_\tau^s x_\tau^s \le a_t^s \qquad \forall t, s \tag{2}$$

$$x_t^s = \tilde{x}_t^{s,s'} \quad \forall t \in T_I, \forall (s,s') \in P_3 \tag{3b}$$

$$-M(1 - z_t^{s,s'}) \quad \le x_t^s - \tilde{x}_t^{s,s'} \quad \forall t \in T_C, \forall (s,s') \in P_3 \tag{5d}$$

$$M(1 - z_t^{s,s'}) \quad \ge x_t^s - \tilde{x}_t^{s,s'} \quad \forall t \in T_C, \forall (s,s') \in P_3 \tag{5e}$$

$$B_t^s x_t^s + C_t^s z_t^{s,s'} \le d_t^s \qquad \forall t \in T_C, \forall (s,s') \in P_3 \tag{4b}$$

$$x_{jt}^s \in \{0,1\} \qquad \forall t, s, \forall j \in J' \tag{6}$$

$$x_{jt}^s \in R \qquad \forall t, s, \forall j \in J \setminus J' \tag{7}$$

(L3-MLR^{Cs})

$$\min \quad \sum_{t \in T} p^s c_t x_t^s + \sum_{t \in T} [x_t^s \sum_{\substack{(s',s) \in P_3 \\ s>s'}} \lambda_t^{s',s} - \sum_{\substack{(s,s') \in P_3 \\ s<s'}} \lambda_t^{s,s'} \tilde{x}_t^{s,s'}] \tag{1f}$$

$$s.t. \quad \sum_{\tau \le t} A_\tau^s x_\tau^s \le a_t^s \qquad \forall t \tag{2a}$$

$$x_t^s = \tilde{x}_t^{s,s'} \quad \forall t \in T_I, \forall (s,s') \in P_3, s < s' \tag{3c}$$

$$-M(1 - z_t^{s,s'}) \quad \le x_t^s - \tilde{x}_t^{s,s'} \quad \forall t \in T_C, \forall (s,s') \in P_3, s < s' \tag{5g}$$

$$M(1 - z_t^{s,s'}) \quad \ge x_t^s - \tilde{x}_t^{s,s'} \quad \forall t \in T_C, \forall (s,s') \in P_3, s < s' \tag{5h}$$

$$B_t^s x_t^s + C_t^s z_t^{s,s'} \le d_t^s \qquad \forall t \in T_C, \forall (s,s') \in P_3, s < s' \tag{4c}$$

$$x_{jt}^s \in \{0,1\} \qquad \forall t, s, \forall j \in J' \tag{6a}$$

$$x_{jt}^s \in R \qquad \forall t, s, \forall j \in J \setminus J' \tag{7a}$$

Notice that once the scenario subproblems L2-MLR^s and L3-MLR^{Cs} corresponding to the direct and indirect approaches, (i) and (ii), are formulated, the rest of the algorithmic steps are similar to as we have seen in the previous section (Figure 2).

4.2.2 Limitations: Based on the computational experiments, approach (ii) performs slightly better than the approach (i). However, the main limitation with both of these decomposition approaches (i) and (ii) is that the number of Lagrangean multipliers becomes very large since the conditional NACs represent a very large fraction of the total constraints in the problem. In addition, these constraints appear as big-M constraints in the model where only a small fraction of these constraints become active at the optimal solution, so the improvement in the resulting lower bound is usually very slow and one may need several iterations to converge. Overall, the performance with the decomposition approaches that rely on considering all the conditional NACs can even be worse than the decomposition approach presented in section 4.1.1 which relaxes all of these constraints.

However, for the problems with exogenous uncertainties, there is no big-M involved in the NACs. Therefore, on dualizing these NACs (all time periods) for scenario decomposition, the quality of the lower bound is usually strengthened.

5 Proposed Lagrangean Decomposition Algorithm

The decomposition approaches presented in the previous section may perform reasonably well for a certain class of problems with a given set of data. However, as we mentioned these methods also have some limitations. To overcome them, we propose a new decomposition scheme that neither relaxes nor dualizes all the conditional NACs. The basic idea relies on decomposing the fullspace model into scenario group subproblems instead of individual scenarios. This allows keeping a subset of the NACs in the subproblems as constraints, while dualizing and relaxing the rest of the NACs. Therefore, it can be considered as a partial decomposition approach. Since, the formulation of the scenario groups is a key element in the proposed decomposition algorithm, we first describe the methodology to construct these scenario groups for the MSSP with endogenous uncertainties.

5.1.1 Formulating the Scenario Groups: The proposed algorithm divides the reduced model (MLR) into scenario group subproblems as explained in this section. Let us consider that there are two endogenous uncertain parameters $\{\theta_1, \theta_2\}$ where each one has 2 possible realizations (L, H). Therefore, there are 4 scenarios (1: LL, 2: HL, 3: LH, 4: HH). The scenario pairs *(s,s')* required to represent the NA constraints in each time period *t* based on the three properties in Gupta and Grossmann (2011) are {(1,2),(1,3),(2,4),(3,4)} as can be seen in Figure 7(a). Notice that the double line between scenario pairs is used to emphasize the fact that there are initial as well as conditional NACs between each of these scenario pairs, whereas each node represents the index of an individual scenario.

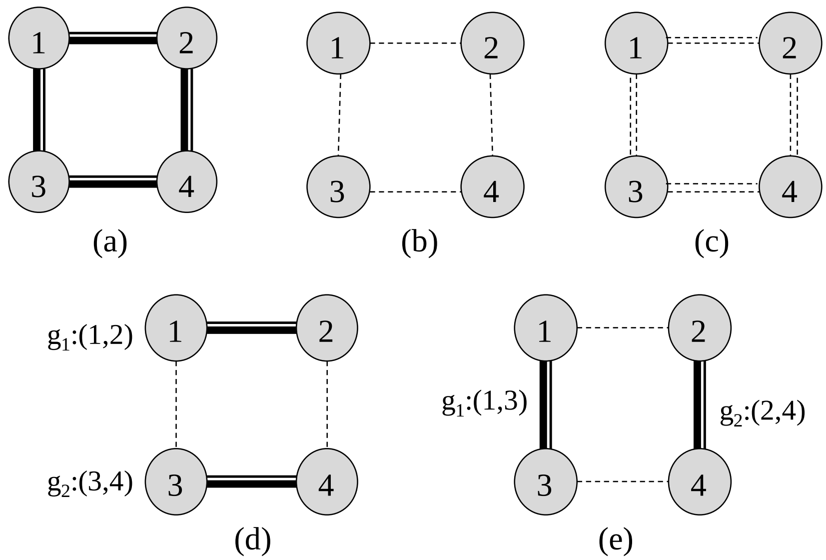

Figure 7: An illustration for the 4 Scenarios and its scenario group decomposition (top view)

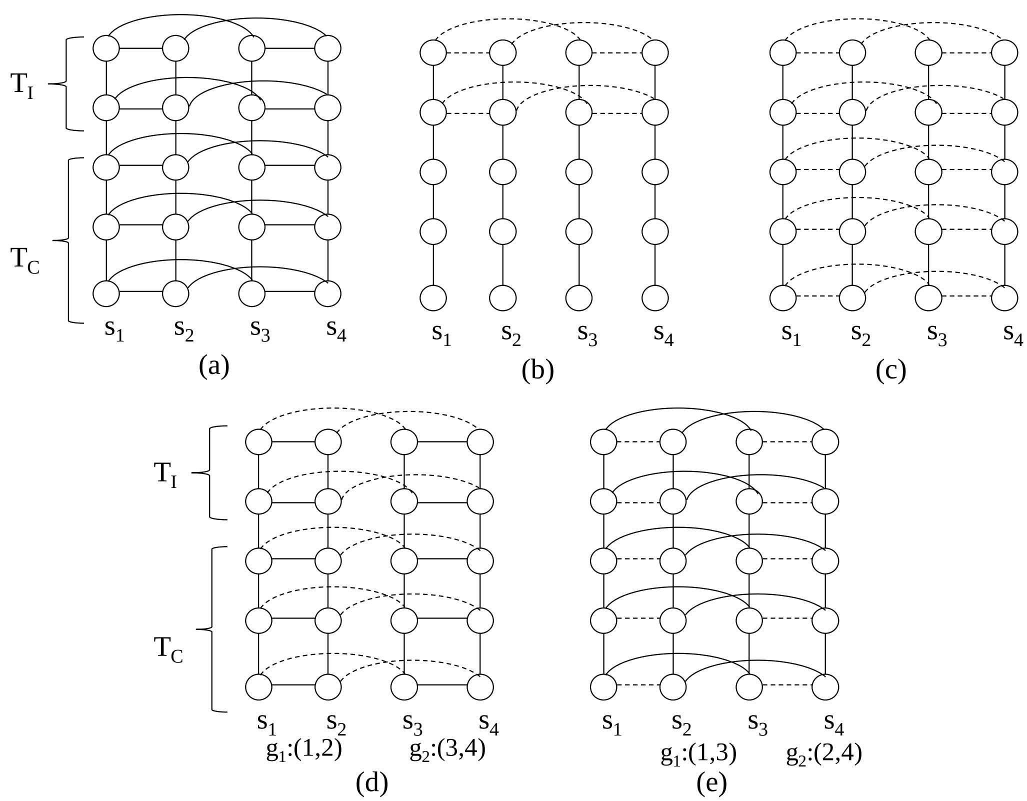

Figure 8: An illustration for the 4 Scenarios and its scenario group decomposition (front view)

The Lagrangean decomposition scheme corresponding to the section 4.1.1 is represented by Figure 7(b) where we remove all the conditional NACs and dualize all the initial NACs. Figure 7(c) corresponds to the scenario decomposition scheme presented in section 4.2.1 that relies on dualizing all the NACs (initial and conditional) either directly (i) or after reformulation (ii). In contrast, the proposed algorithm decomposes the fullspace model into scenario groups as shown in Figures 7(d) or 7(e). In particular, Figure 7(d) corresponds to the two scenario group problems {g_1: (1,2), g_2: (3,4)} where 7(e) represents the scenario group problems { g_1: (1,3), g_2: (2,4)}. Notice that Figures 7(a)-(e) correspond to the top view of the scenario-tree representation in Figures 8(a)-(e), respectively. Each node in Figures 8(a-e) represents the state of the system in a given time period *t* while the linking lines correspond to the NA constraints.

The rules to formulate the scenario groups for the proposed algorithm are as follows:

1. Each scenario *s* occurs in only one of the scenario group S_g and every scenario is included in at-least one of the groups. All the scenario groups $S_g \in G$ have equal number of scenarios. Therefore, the total number of scenarios equal to the number of scenario groups times the number of scenarios in each group i.e., $|S| = |G| \cdot |S_g|$. Notice that here we assume the symmetry of the scenario groups to formulate the subproblems that have almost similar complexity. However, we can always consider an asymmetric approach as shown in Figure 9 for the 4 scenario instance described above. Specifically, Figure 9(a) and 9(b) decompose the problem into two scenario groups {g_1: (1,2,3), g_2: (4)} and {g_1: (1,3,4), g_2: (2)}, respectively, where the subproblems with 3 scenarios should be more expensive to solve than the one with a single scenario.

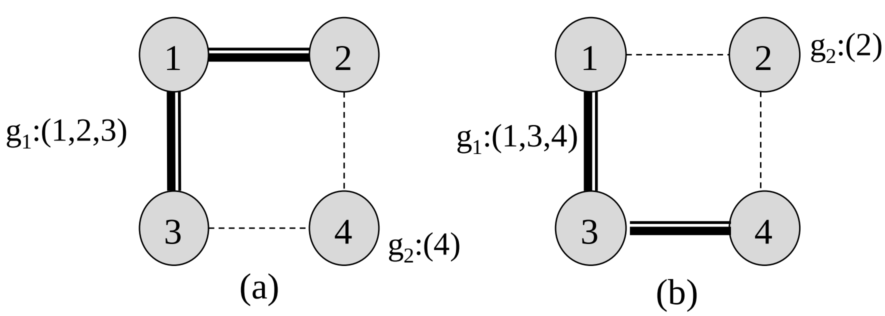

Figure 9: Asymmetric scenario group decomposition

2. Scenario groups S_g are formulated by first selecting an endogenous uncertain parameter and then taking those scenarios in a group which differ in the realization of only that particular

uncertain parameter. For instance, in Figure 7(a), we first select parameter $\{\theta_1\}$ and write only those scenario groups that differ in the realization of this uncertain parameter, i.e. $\{(1,2),(3,4)\}$ which results in the scenario groups as in Figure 7(d). Similarly, the uncertain parameter $\{\theta_2\}$ leads to the scenario groups $\{(1,3),(2,4)\}$ in Figure 7(e). Notice that these scenario groups are nothing but the scenario sets $(s_1, s_2, ..., s_k) \in L_p$ (eq. 8) that are required to formulate the reduced model (MLR).

3. Since there can be many uncertain parameters $\{\theta_p\}$ each with its own scenario set $(s_1, s_2, ..., s_k) \in L_p$, the selection of a particular set of scenario groups is not unique.
   (i) Ideally, one may consider selecting a scenario group set that provides the tightest initial bound compared to the others. However, in general unless all the combinations are tested, it is not obvious how to select such a scenario group set.
   (ii) A relatively simpler approach can be to first solve each scenario independently, and selecting the scenario group set corresponding to that uncertain parameter, which has the largest total difference in the objective function values of the corresponding scenarios. This is due to the fact that most likely the corresponding NACs for those scenarios will be active at the optimal solution. Therefore, keeping these NACs in the subproblem as constraints should yield a tighter bound. For instance, select 7(e) if scenario group set corresponding to θ_2 exhibits larger total variation in the objective function value than the scenario group set for uncertain parameter θ_1. In other words, this idea relies on the sensitivity of the objective function value for an uncertain parameter and its possible realizations.
4. Even after selecting a scenario group set that corresponds to an uncertain parameter $\{\theta_p\}$, it may still be difficult to solve the resulting scenario group subproblems. For instance if a parameter has many realizations, then each scenario group subproblem will have that many scenarios which may increase the computational expense. Therefore, one may further divide the scenario groups into subgroups and solve the resulting smaller problems.

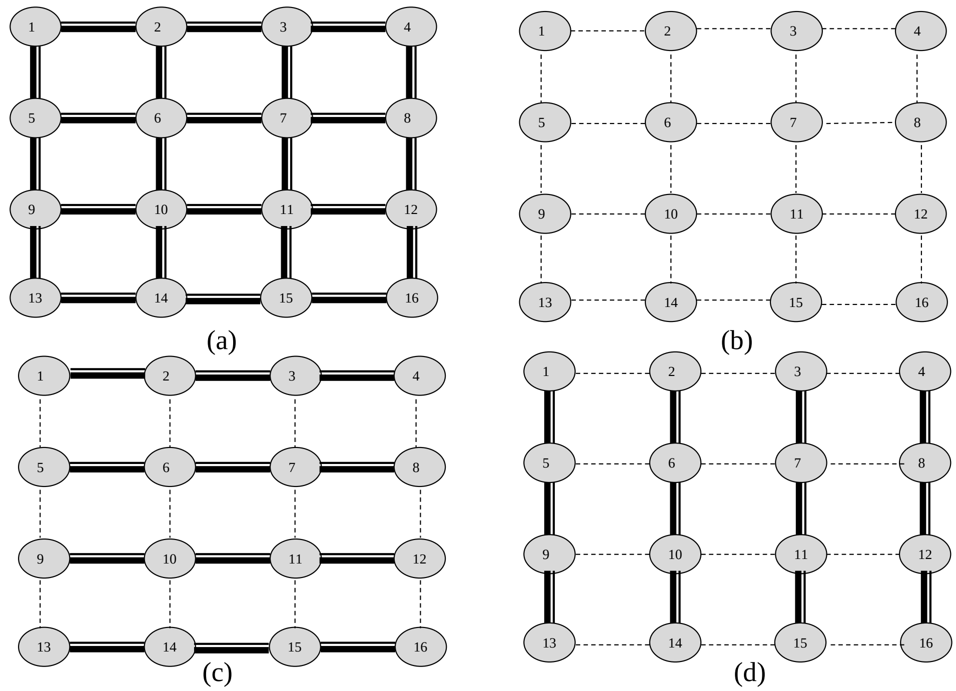

Figure 10: 2 parameters, 16 scenarios and its scenario/scenario group decomposition

As an example, say if we have 2 uncertain parameters and 4 realizations of each parameter, there are a total of 16 scenarios. There are two possibilities of the scenario groups {(1,2,3,4), (5,6,7,8), (9,10,11,12), (13,14,15,16)} (Figure 10(c)) and {(1,5,9,13), (2,6,10,14), (3,7,11,15), (4,8,12,16)} (Figure 10 (d)) according to the rules 1-3. Based on the problem characteristics, it may be difficult to solve each scenario group subproblem with 4 scenarios. Therefore, these groups can be further decomposed into a total of 8 scenario groups each with 2 scenarios, respectively (Figure 11(a) and 11(b)). However, the quality of the bound may deteriorate since the corresponding conditional NACs need to be relaxed. Therefore, there is a trade-off between the quality of the bound and the complexity of solving a scenario group problem.

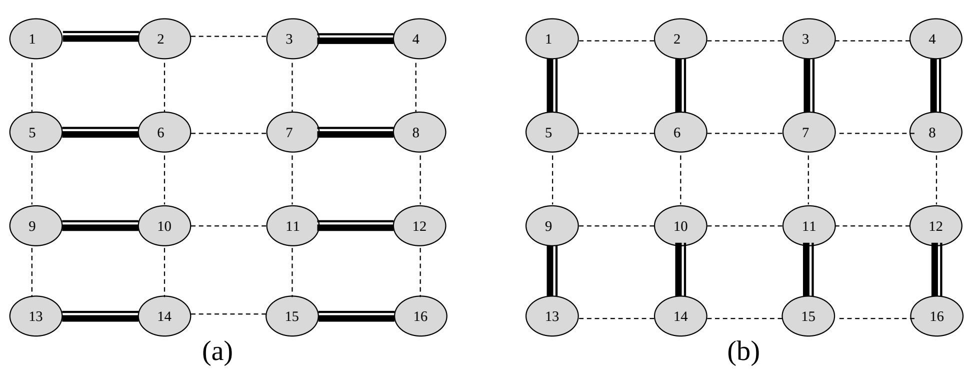

Figure 11: Decomposition of the scenario groups into subgroups

5. In general, if the problem is expensive to solve for each scenario, it is better to use scenario groups each with only few scenarios. On the other hand, if individual scenarios are not expensive to solve, then one may consider more scenarios in each group to improve the quality of the bound.

The above rules are generic and can be applied to a problem with any number of uncertain parameters and many realizations of each uncertain parameter. For instance, Figure 12(a) represents the extension to three uncertain parameter case where each parameter has 2 realizations (total 8 scenarios).

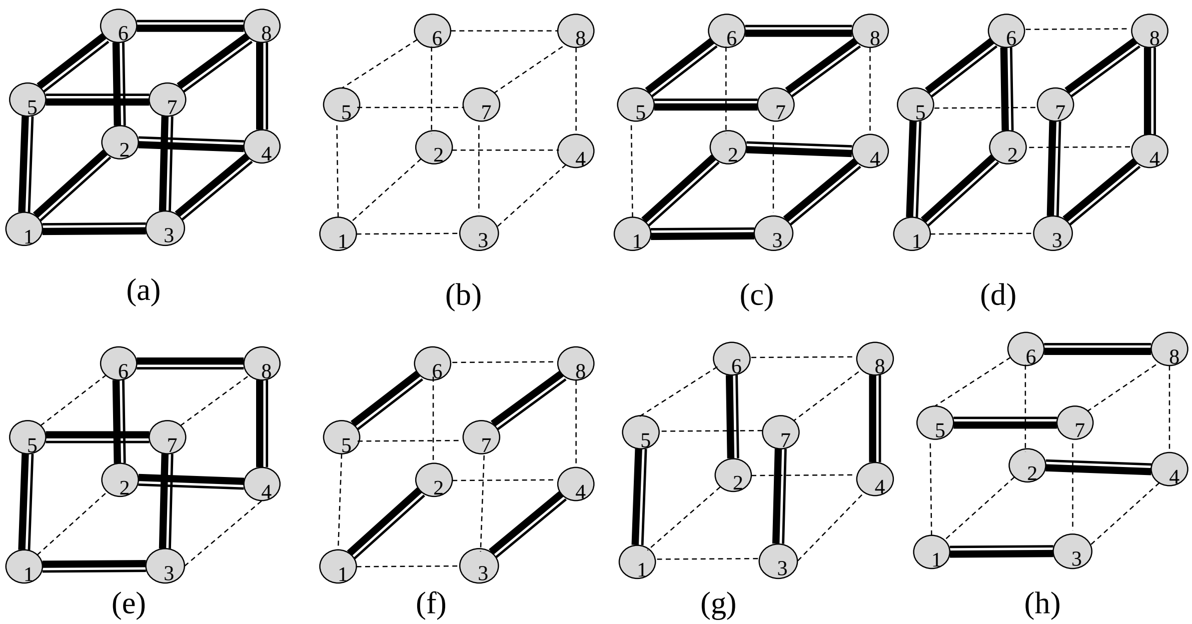

Figure 12: 3 parameters, 8 scenarios and its scenario/scenario group decomposition

There are 6 possibilities to formulate the scenario groups in symmetric form:

(a) Taking 4 scenarios in each group: {(1,2,3,4), (5,6,7,8)} i.e. Figure 12(c)
{(1,2,5,6), (3,4,7,8)} i.e. Figure 12(d)
{(1,3,5,7), (2,4,6,8)} i.e. Figure 12(e)

(b) Taking 2 scenarios in each group: {(1,2),(3,4), (5,6),(7,8)} i.e. Figure 12(f)
{(1,5),(2,6), (3,7),(4,8)} i.e. Figure 12(g)
{(1,3),(2,4), (5,7),(6,8)} i.e. Figure 12(h)

5.1.2 Decomposition Algorithm: Based on the scenario groups that are constructed in the previous section, we now first present the corresponding reformulated Reduced (MILP) model. Notice that these scenario group partitions will be used to decompose the resulting reduced model into scenario group subproblems during the proposed Lagrangean decomposition algorithm.

Let us consider that G is the set of scenario groups $S_g \in G$ that are selected based on the rules presented in the previous section, where each of these scenario groups S_g may have 1 or more scenarios. The reduced model (MLR) can now be represented as an equivalent model (**MLR^{G}**) in terms of the scenario groups $S_g \in G$ where we disaggregate the total NACs for the scenario pairs that corresponds to the same scenario group $(s,s') \in S_g$ (i.e. eqs. (3i), (4i),(5i),(5j)) with those which belong to the different scenario groups $(s \in S_g) \wedge (s' \notin S_g)$ (i.e. eqs. (3j), (4j),(5k),(5l)).

(MLR^{G})

$$\min \sum_{S_g \in G} \left\{ \sum_{s \in S_g} p^s \sum_{t \in T} c_t x_t^s \right\} \tag{1i}$$

$$s.t. \quad \sum_{\tau \le t} A_\tau^s x_\tau^s \le a_t^s \qquad \forall t, s \in S_g \in G \tag{2i}$$

$$x_t^s = x_t^{s'} \quad \forall t \in T_I, \forall (s,s') \in P_3, \; s,s' \in S_g \in G \tag{3i}$$

$$B_t^s x_t^s + C_t^s z_t^{s,s'} \le d_t^s \qquad \forall t \in T_C, \forall (s,s') \in P_3, s,s' \in S_g \in G \tag{4i}$$

$$-M(1 - z_t^{s,s'}) \quad \le x_t^s - x_t^{s'} \quad \forall t \in T_C, \forall (s,s') \in P_3, s,s' \in S_g \in G \tag{5i}$$

$$M(1 - z_t^{s,s'}) \quad \ge x_t^s - x_t^{s'} \quad \forall t \in T_C, \forall (s,s') \in P_3, s,s' \in S_g \in G \tag{5j}$$

$$x_t^s = x_t^{s'} \quad \forall t \in T_I, \forall (s,s') \in P_3, (s \in S_g) \wedge (s' \notin S_g), S_g \in G \tag{3j}$$

$$B_t^s x_t^s + C_t^s z_t^{s,s'} \le d_t^s \qquad \forall t \in T_C, \forall (s,s') \in P_3, (s \in S_g) \wedge (s' \notin S_g), S_g \in G \tag{4j}$$

$$-M(1 - z_t^{s,s'}) \quad \le x_t^s - x_t^{s'} \quad \forall t \in T_C, \forall (s,s') \in P_3, (s \in S_g) \wedge (s' \notin S_g), S_g \in G \tag{5k}$$

$$M(1 - z_t^{s,s'}) \quad \ge x_t^s - x_t^{s'} \quad \forall t \in T_C, \forall (s,s') \in P_3, (s \in S_g) \wedge (s' \notin S_g), S_g \in G \tag{5l}$$

$$x_{jt}^s \in \{0,1\} \qquad \forall t, \forall s \in S_g \in G, \forall j \in J' \tag{6i}$$

$$x_{jt}^{s} \in R \qquad \forall t, \forall s \in S_g \in G, \forall j \in J \setminus J' \tag{7i}$$

The Lagrangean problem (**L4-MLR^{G}**) corresponding to the model (MLR^{G}) can be formulated by dualizing only those initial NAC constraints for the pairs of scenarios *(s,s')* that link the two scenario groups, i.e. eq. (3j), and removing the corresponding conditional NACs (eqs. (4j),(5k) and (5l)). Therefore, the initial and conditional NACs (eq. 3(i), 4(i), 5(i) and 5(j)) among the scenario pairs *(s,s')* that belong to the same scenario group remain in the Lagrangean problem as explicit constraints.

(**L4-MLR^{G}**)

$$\min \sum_{S_g \in G} \left\{ \sum_{s \in S_g} p^s \sum_{t \in T} c_t x_t^s + \sum_{t \in T_1} \sum_{\substack{(s,s') \in P_3 \\ s.t.\ (s \in S_g) \wedge (s' \notin S_g)}} \lambda_t^{s,s'} (x_t^s - x_t^{s'}) \right\} \tag{1j}$$

$$s.t. \quad \sum_{\tau \le t} A_\tau^s x_\tau^s \le a_t^s \qquad \forall t, s \in S_g \in G \tag{2i}$$

$$x_t^s = x_t^{s'} \quad \forall t \in T_I, \forall (s,s') \in P_3,\ s,s' \in S_g \in G \tag{3i}$$

$$B_t^s x_t^s + C_t^s z_t^{s,s'} \le d_t^s \qquad \forall t \in T_C, \forall (s,s') \in P_3, s,s' \in S_g \in G \tag{4i}$$

$$-M(1 - z_t^{s,s'}) \quad \le x_t^s - x_t^{s'} \quad \forall t \in T_C, \forall (s,s') \in P_3, s,s' \in S_g \in G \tag{5i}$$

$$M(1 - z_t^{s,s'}) \quad \ge x_t^s - x_t^{s'} \quad \forall t \in T_C, \forall (s,s') \in P_3, s,s' \in S_g \in G \tag{5j}$$

$$x_{jt}^{s} \in \{0,1\} \qquad \forall t, \forall s \in S_g \in G, \forall j \in J' \tag{6i}$$

$$x_{jt}^{s} \in R \qquad \forall t, \forall s \in S_g \in G, \forall j \in J \setminus J' \tag{7i}$$

In contrast to the previous approaches, we can observe that the main idea in the proposed decomposition approach is that instead of removing all the conditional NACs from the model (as in section 4.1.1) or dualizing all the conditional NACs either directly or in an indirect manner (as in section 4.2.1), we only remove a subset of conditional NACs from the model and dualize a subset of the initial NACs in the objective function instead of dualizing all the initial NACs while formulating the Lagrangean problem (L4-MLR^{G}). This results in the decomposition of the reduced model (MLR) into scenario group subproblems (**L4-MLR^{Gs}**) rather than individual scenarios in the previous cases. Therefore, we also refer it as a partial decomposition approach.

$$\textbf{(L4-MLR}^{\textbf{Gs}}\textbf{)} \qquad \min \sum_{s \in S_g} p^s \sum_{t \in T} c_t x_t^s + \sum_{t \in T_1} x_t^s \Big(\sum_{\substack{(s,s') \in P_3, s<s' \\ s.t. (s \in S_g) \wedge (s' \notin S_g)}} \lambda_t^{s,s'} - \sum_{\substack{(s',s) \in P_3, s>s' \\ s.t. (s \in S_g) \wedge (s' \notin S_g)}} \lambda_t^{s',s} \Big) \tag{1k}$$

$$s.t. \quad \sum_{\tau \le t} A_\tau^s x_\tau^s \le a_t^s \qquad \forall t, s \in S_g \tag{2k}$$

$$x_t^s = x_t^{s'} \quad \forall t \in T_I, \forall (s,s') \in P_3,\ s,s' \in S_g \tag{3k}$$

$$B_t^s x_t^s + C_t^s z_t^{s,s'} \le d_t^s \qquad \forall t \in T_C, \forall (s,s') \in P_3, s,s' \in S_g \tag{4k}$$

$$-M(1 - z_t^{s,s'}) \le x_t^s - x_t^{s'} \quad \forall t \in T_C, \forall (s,s') \in P_3, s,s' \in S_g \tag{5m}$$

$$M(1 - z_t^{s,s'}) \ge x_t^s - x_t^{s'} \quad \forall t \in T_C, \forall (s,s') \in P_3, s,s' \in S_g \tag{5n}$$

$$x_{jt}^s \in \{0,1\} \qquad \forall t, \forall s \in S_g, \forall j \in J' \tag{6k}$$

$$x_{jt}^s \in R \qquad \forall t, \forall s \in S_g, \forall j \in J \setminus J' \tag{7k}$$

The structure of model (L4-MLR^G) can be seen in Figure 13, where each scenario group subproblem that contains its corresponding initial and conditional NACs can be solved independently, and where only a small fraction of the total initial and conditional NACs are dualized and removed, respectively. Since, the resulting subproblems capture the more relevant information, i.e. the one corresponding to the later time periods, the dual bound should be tighter. We can then state the following proposition:

Proposition 1: ***The dual bound obtained from the proposed Lagrangean problem (L4-MLR^G) at root node is at-least as tight as the dual bound obtained from the standard Lagrangean decomposition approach (L1-MLR) i.e. the model (L1-MLR) is a relaxation of the model (L4-MLR^G).***

Proof: To prove this proposition it is sufficient to establish that,

(a) The feasible region of the proposed Lagrangean problem (L4-MLR^G) is contained within the feasible region of the model L1-MLR.

(b) The objective function value of the proposed Lagrangean problem (L4-MLR^G) over its feasible solutions x_t^s is at-least as large (assuming minimization case) as the objective function value of the model L1-MLR.

For (a), since scenario constraints (2) in L1-MLR are equivalent to constraints (2i) in L4-MLR^G. Therefore, the only difference between both of these models is that L4-MLR^G has the additional constraints (3i), (4i), (5i) and (5j) in the model. Hence, the feasible region of the model L4-MLR^G is contained within the feasible region of the standard Lagrangean problem L1-MLR which has more feasible solutions.

For (b), we first rewrite the model L1-MLR as **L1-MLR'** where $\eta_t^{s,s'} \geq 0$ represent the Lagrangean multipliers corresponding to the dualized inequalities $(x_t^s - x_t^{s'} \leq 0)$ and multipliers $\mu_t^{s,s'} \geq 0$ correspond to the inequalities $(-x_t^s + x_t^{s'} \leq 0)$. We use the inequality format of the initial NACs (eq. (3a)) to dualize them in the objective function.

(L1-MLR')

$$\min \quad \sum_{s \in S} p^s \sum_{t \in T} c_t x_t^s + \sum_{t \in T_1} \sum_{(s,s') \in P_3} \eta_t^{s,s'} (x_t^s - x_t^{s'}) + \sum_{t \in T_1} \sum_{(s,s') \in P_3} \mu_t^{s,s'} (-x_t^s + x_t^{s'}) \tag{1l}$$

s.t. (2), (6) and (7)

Similarly, model L4-MLR^G can be rewritten as follows:

(L4-MLR$^{G'}$)

$$\min \quad \sum_{S_g \in G} \left\{ \sum_{s \in S_g} p^s \sum_{t \in T} c_t x_t^s + \sum_{t \in T_1} \sum_{\substack{(s,s') \in P_3 \\ s.t. (s \in S_g) \wedge (s' \notin S_g)}} \eta_t^{s,s'} (x_t^s - x_t^{s'}) + \sum_{t \in T_1} \sum_{\substack{(s,s') \in P_3 \\ s.t. (s \in S_g) \wedge (s' \notin S_g)}} \mu_t^{s,s'} (-x_t^s + x_t^{s'}) \right\} \tag{1m}$$

s.t. (2i), (3i), (4i), (5i), (5j), (6i) and (7i)

On subtracting the objective functions (1l) and (1m), we have the following summation,

$$\sum_{S_g \in G} \left\{ \sum_{t \in T_1} \sum_{\substack{(s,s') \in P_3 \\ s.t. (s \in S_g) \wedge (s' \in S_g)}} \eta_t^{s,s'} (x_t^s - x_t^{s'}) + \sum_{t \in T_1} \sum_{\substack{(s,s') \in P_3 \\ s.t. (s \in S_g) \wedge (s' \in S_g)}} \mu_t^{s,s'} (-x_t^s + x_t^{s'}) \right\} \tag{1n}$$

To prove that the objective function value of the model L4-MLR^G over its feasible solutions x_t^s is at least as large as the objective function value of the model L1-MLR, it is sufficient to prove that,

$$\sum_{S_g \in G} \left\{ \sum_{t \in T_1} \sum_{\substack{(s,s') \in P_3 \\ s.t.\,(s \in S_g) \wedge (s' \in S_g)}} \eta_t^{s,s'} (x_t^s - x_t^{s'}) + \sum_{t \in T_1} \sum_{\substack{(s,s') \in P_3 \\ s.t.\,(s \in S_g) \wedge (s' \in S_g)}} \mu_t^{s,s'} (-x_t^s + x_t^{s'}) \right\} \le 0 \qquad (1o)$$

For any feasible solution x_t^s to the model L4-MLR^G and for any $\eta_t^{s,s'} \ge 0$ and $\mu_t^{s,s'} \ge 0$ $\forall (s,s') \in P_3, t \in T$, the penalty terms $\eta_t^{s,s'}(x_t^s - x_t^{s'})$ and $\mu_t^{s,s'}(-x_t^s + x_t^{s'})$ in the objective function are less than or equal to zero. Hence, their summation in inequality (1o) also holds true. In other words, we can also state that the model L1-MLR is a Lagrangean relaxation of the model L4-MLR^G and therefore, it provides a valid lower bound on the objective function value of the model L4-MLR^G. □

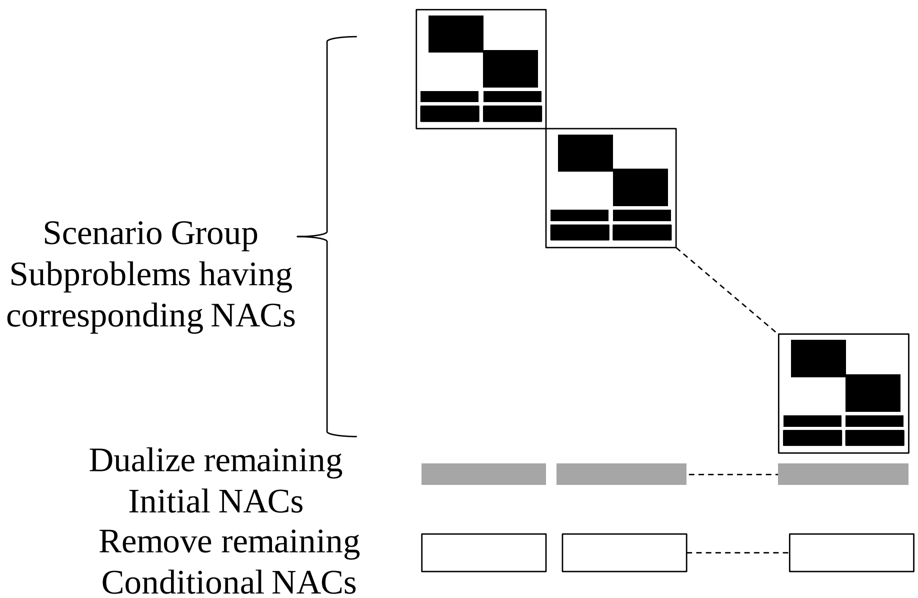

Figure 13: Scenario decomposition approach in the proposed Lagrangean Decomposition

The rest of the steps of the algorithm are similar to the standard Lagrangean decomposition (Figure 2) where scenario group subproblems L4-MLR^{Gs} are solved during each iteration, and multipliers are updated using either subgradient method (Fisher, 1985) or an alternative scheme as in Mouret et al. (2011); Oliveira et al. (2013), and Tarhan et al. (2013). Moreover, the algorithm can be further extended within a duality based branch and bound search (as proposed in Goel and Grossmann, 2006; Tarhan et al., 2009; and Tarhan et al., 2011) if the gap between the lower and upper bound is still large. As will be shown in the results, the main advantage with the proposed approach is that the resulting dual bound is significantly strengthened at the root node itself since a large fraction of the NACs are included as explicit constraints in the subproblems. This will eventually reduce the number of iterations required to converge at each node and the total number of nodes in the branch and bound search.

5.2.1 Alternate Proposed Lagrangean Decomposition Algorithm

It should be noted that few conditional NACs (eqs. (5k) and (5l)) still need to be removed while formulating the scenario group subproblems (L4-MLR^{Gs}) in the above method. Therefore, the best lower bound at the root node cannot be better than the optimal solution of the model without these conditional NACs. To further close the gap at the root node, we also propose an alternate Lagrangean decomposition approach that may provide a stronger bound at the root node. However, it involves solving more subproblems, and it may be computationally more expensive than the proposed approach in the previous section. Therefore, it is only useful for a certain class of problems.

The main idea is that we select all the scenario groups instead of a subset of the scenario groups as we did in the previous section 5.1.1. However, since a scenario can appear in more than one of these scenario groups, we need to equate the decisions for this scenario in all of these scenario groups where it occurs. In other words, we create a copy of each scenario for every scenario group problem where it can appear and equating the decisions corresponding to all time periods for that scenario for each of these scenario groups. The resulting model (Figure 14(b)) will be equivalent to the reduced model (MLR) (Figure 14(a)) where {1',2',3',4'} are the copy of the scenarios {1,2,3,4} and the connections between them are the added equality constraints.

Therefore, to decompose the resulting problem (Figure 14(b)) into 4 scenario group subproblems {(1,2),(1',3'),(2',4'), (3,4)}, we dualize the equality constraints correspond to each scenario and its copy variables, instead of dualizing or removing the NAC constraints. This yields a set of 4 scenario group subproblems (Figure 14(c)) i.e. {(1,2),(1',3'),(2',4'), (3,4)}. Since, none of the conditional and initial NAC constraints are removed from the subproblems, the bound is in general stronger. We can compare this decomposition with the proposed one in Figure 7 where we obtain 2 scenario group problems.

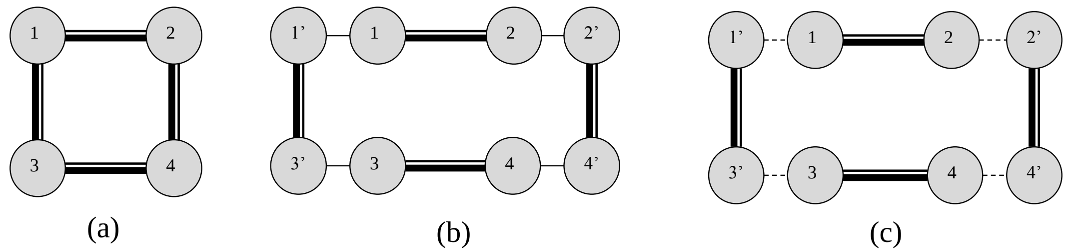

Figure 14: Alternate proposed Lagrangean decomposition approach for 4 scenario problem

Qualitatively, this decomposition can be considered as the decomposition of the reduced model (Figure 14(a)) at vertices as compared to the arcs in standard/proposed decomposition described earlier. Notice that although this alternate decomposition is computationally expensive since more subproblems are involved than in the previous method, it can however be used in a hybrid scheme with the proposed decomposition to improve the quality of lower bound. For instance in Figure 10, we can first select the 4 scenario groups based on the rules that are defined earlier, and then use this approach to further decompose each group into subgroups by creating a copy of the scenarios in each of these groups instead of the partitions used in Figure 11.

6. Numerical Results

6.1 Process network planning under uncertain yield

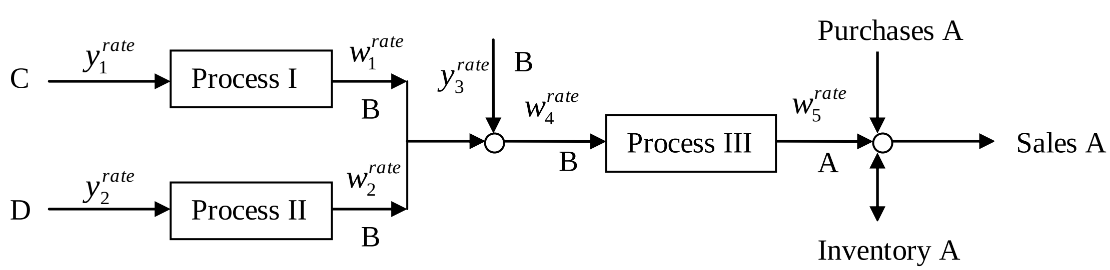

Figure 15: 3 Process Network Example

Case (i): Planning of 3 process network over 10 years

To illustrate the application of the various decomposition approaches for multistage stochastic programming with endogenous uncertainties, we consider the following problem from Goel and Grossmann (2006). Given is a process network (Figure 15) that is used to produce product A. Currently, the production of A takes place only in Process III with installed capacity of 3 tons/hour and yield of 0.70, that consumes an intermediate product B which is purchased. If needed, the final product A can also be purchased so as to maintain its inventory. The demand for the final product, which is known, must be satisfied for all time periods over the given time horizon. Two new technologies (Process I and Process II) are considered for producing the intermediate B from two different raw materials C and D. These new technologies exhibit uncertainty in the yields. The yield of Process I and Process II can take 2 discrete values each with equal probability of 0.5. These two realizations of yield for each of Process I and Process II give rise to a total of 4 scenarios (Table 3).

The problem consists of finding the expansion and operation decisions for this process network for a 10 year planning horizon so as to minimize the total expected cost of the project. The size of the resulting fullspace model (MLR) and each individual scenario can be seen in Table 4 where the optimal expected cost of the problem is \$379,070. Notice that there is a significant increase in the total number of constraints for the fullspace MSSP model due to the non-anticipativity requirements.

Table 3: 3 Process Network Example (4 Scenarios)

Scenario	s1	s2	s3	s4
Process I yield	0.69	0.81	0.69	0.81
Process II yield	0.65	0.65	0.85	0.85
Scenario Probability	0.25	0.25	0.25	0.25

Table 4: Model statistics for the 3 Process Network Example

Problem Type	Number of Constraints	Continuous Variables	Binary Variables
Reduced Model (MLR)	1,869	845	120
Individual Scenario	192	202	30

After applying the various decomposition approaches, we obtain the results shown in Figure 16 and Table 5, where an optimality tolerance of 1% and maximum of 30 subgradient iterations (whichever comes first) are used as the termination criteria. It can be observed that the proposed approach (section 5.1.2) using SG2 scenario groups {(1,3),(2,4)} outperforms the other approaches since it yields the tightest lower bound (\$378,710) within 2 iterations (see Table 5). The lower bound at the root node from the standard approach (section 4.1.1) after many iterations is worse than the initial bound with the proposed approach (\$375,880 vs. \$377,290). In addition, the best upper bound from the proposed approach is same as the optimal solution (\$379,070) whereas the standard approach could only yield the feasible solution with expected cost of \$380,880 even after 30 iterations (see Table 5). The decomposition approaches based on dualizing all the initial and conditional NACs do not yield good bounds (especially the direct approach (i) in section 4.2) compared to the proposed approach with SG2 partitions.

The alternate decomposition (section 5.2.1) using all the 4 scenario groups also performs reasonably well. Since, the total variations in the scenario costs for the scenario group set SG2 {(1,3),(2,4)} is large compared to the scenario group set SG1 {(1,2),(3,4)} (\$69,990 vs. \$44,590), it yields tighter bounds and faster convergence (see Table 6). Notice that the scenario groups in SG1 represent the sensitivity of the Process I yield with respect to the cost, whereas SG2 correspond to the sensitivity of the Process II yield that has a large variance (Table 3) and a larger impact on the scenario costs. The MILP models for all the process network examples are implemented in GAMS 23.6.3 and run on Intel Core i7, 4GB RAM machine using XPRESS 21.01 solver.

Table 5: Comparison of the various decomposition schemes for 3 Process Network Example

	Standard	All Dualized (i) direct	All Dualized (ii) indirect	Proposed SG1	Proposed SG2	Proposed Alternate
UB ($\$10^3$)	380.88	380.88	380.88	380.88	379.07	379.07
LB ($\$10^3$)	375.88	371.88	376.27	376.42	378.71	375.75
Solution Time (s)	8.89	5.24	9.51	5.86	0.94	2.12
% Gap	1.33%	2.42%	1.22%	1.19%	<1%	<1%
# iterations	30	30	30	30	2	4

Table 6: Variations in the objective function value with uncertain parameters

(a) Individual Scenario Costs

	Cost ($\$10^3$)
s1	410.32
s2	365.73
s3	353.03
s4	353.03

(b) Scenario groups cost variations

	SG1	SG2
s1-s2	44.59	-
s3-s4	0	-
s1-s3	-	57.29
s2-s4	-	12.70
Total cost variations ($\$10^3$)	44.59	69.99

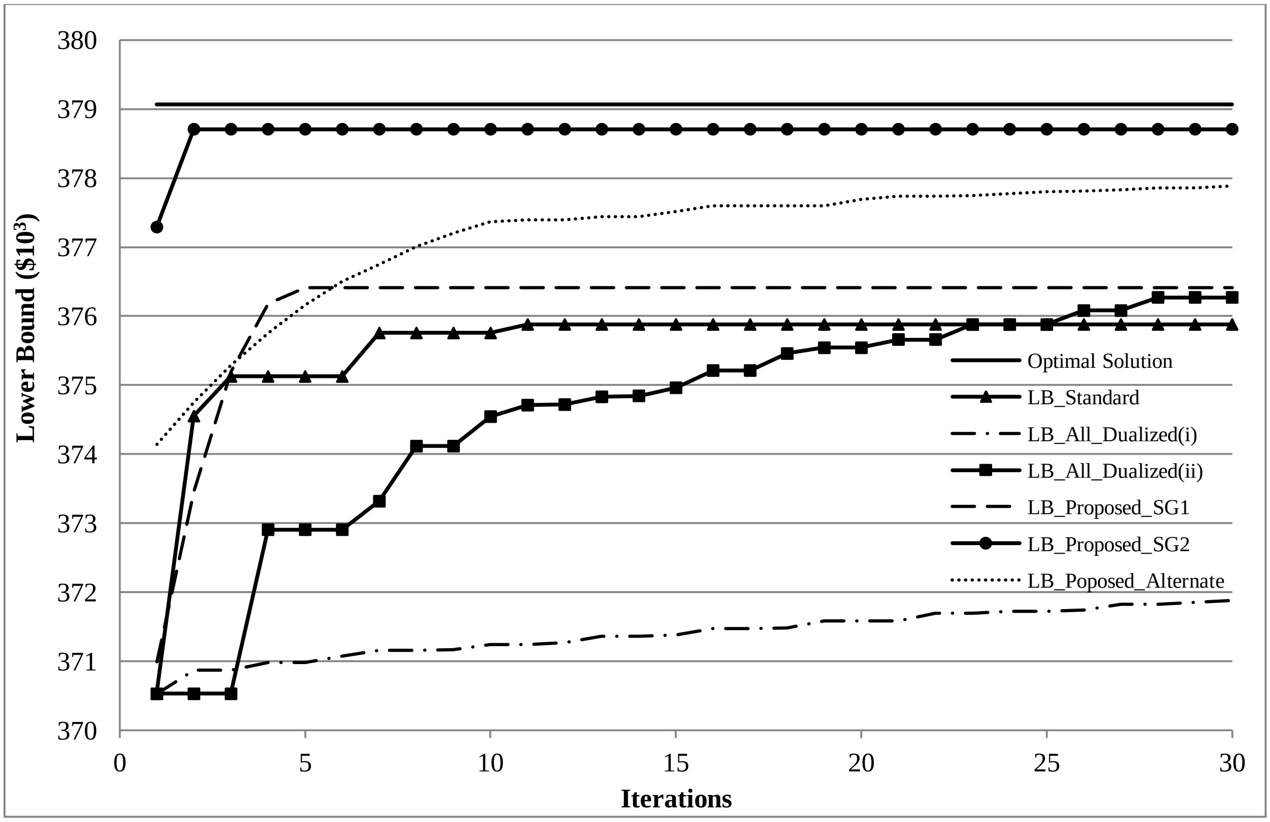

Figure 16: Comparison of the various decomposition schemes for 3 process network example

Case (ii): Planning of 5 process network over 10 years

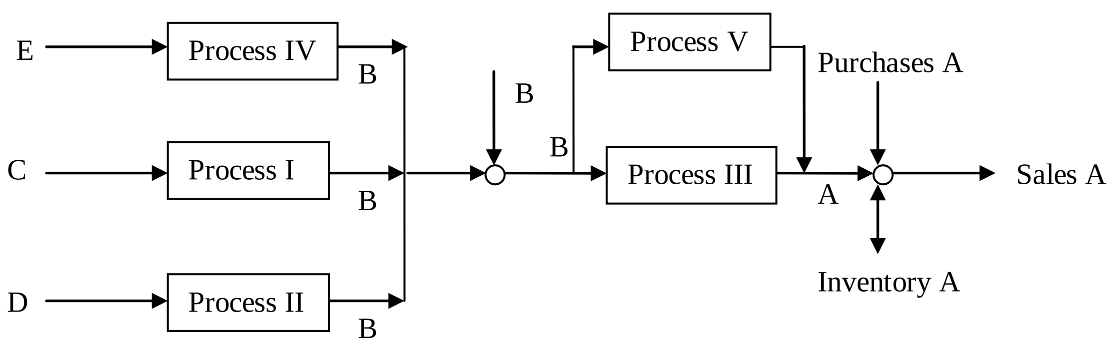

Figure 17: 5 Process Network Example

In this instance, we consider a 5 process network (Figure 17) having 3 uncertain parameters, i.e. yield of Process I, Process II, and Process V. Here we consider 2 new additional processes compared to the previous example in which Process IV converts E into B with a yield of 0.75, and Process V that converts B into final product A. Each of the uncertain yields has 2 realizations and gives rise to a total of 8 scenarios with equal probabilities as shown in Table 7.

The problem consists of finding the expansion and operation decisions for this process network over a 10 year planning horizon to minimize the total expected cost of the project (see Gupta and Grossmann (2011) for details).

Table 7: 5 Process Network Example (4 Scenarios)

Scenario	s1	s2	s3	s4	s5	s6	s7	s8
Process I yield	0.69	0.81	0.69	0.81	0.69	0.81	0.69	0.81
Process II yield	0.65	0.65	0.85	0.85	0.65	0.65	0.85	0.85
Process V yield	0.60	0.60	0.60	0.60	0.80	0.80	0.80	0.80
Scenario Probability	0.125	0.125	0.125	0.125	0.125	0.125	0.125	0.125

To use the proposed decomposition approach for this 8 scenario problem, we partition the scenarios into scenario groups where each one has either 2 or 4 scenarios as in Figure 12. These scenario groups are denoted as follows:

(a) SG1: {(1,2),(3,4), (5,6),(7,8)}; SG2: {(1,5),(2,6), (3,7),(4,8)}; SG3: {(1,3),(2,4), (5,7),(6,8)};
(b) SG4: {(1,2,3,4), (5,6,7,8)}; SG5: {(1,2,5,6), (3,4,7,8)}, and SG6: {(1,3,5,7), (2,4,6,8)}

After applying the proposed decomposition approach (section 5.1.2) to these 6 scenario group sets, we can see from Figure 18 that the quality of the lower bound improves from \$357,920 (SG1) to \$361,500 (SG6) as the total cost variations for the corresponding scenario group set increases from \$41,000 to \$224,810 as in the previous instance. Moreover, the bound obtained from the larger subproblems having 4 scenarios (SG4, SG5, SG6) is tighter as compared to the subproblems having 2 scenario each as in SG1, SG2 and SG3. This is due to the fact that larger subproblems need only few conditional NACs to be relaxed compared to the smaller subproblems. Table 8 and Figure 19 compare the progress of the lower bounds, number of iterations and solution time required to reach within 1% of optimality tolerance (or 30 iterations) for the standard and proposed approaches with different scenario partitions. We can observe that scenario group set SG6 outperforms other approaches since it provides the strongest lower bound (\$361,500) in just 2 iterations within 8.9s. Moreover, there is a trade-off between the computational cost per iteration and the quality of the bound obtained. It is interesting to note that in most of the cases, even the initial bound using proposed scenario decompositions is much better than the final bound from the standard approach (\$355,180) and the rate of convergence to the best possible dual bound is faster.

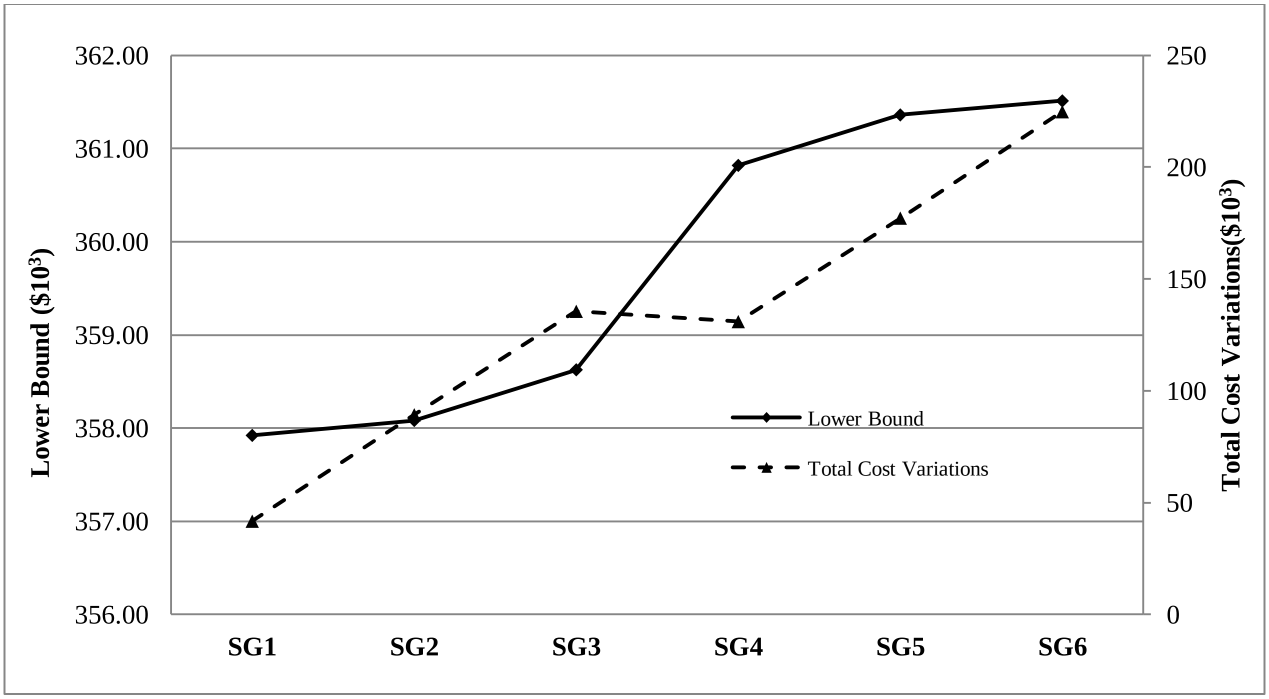

Figure 18: Variations in the scenario costs vs. bound obtained for different scenario partitions

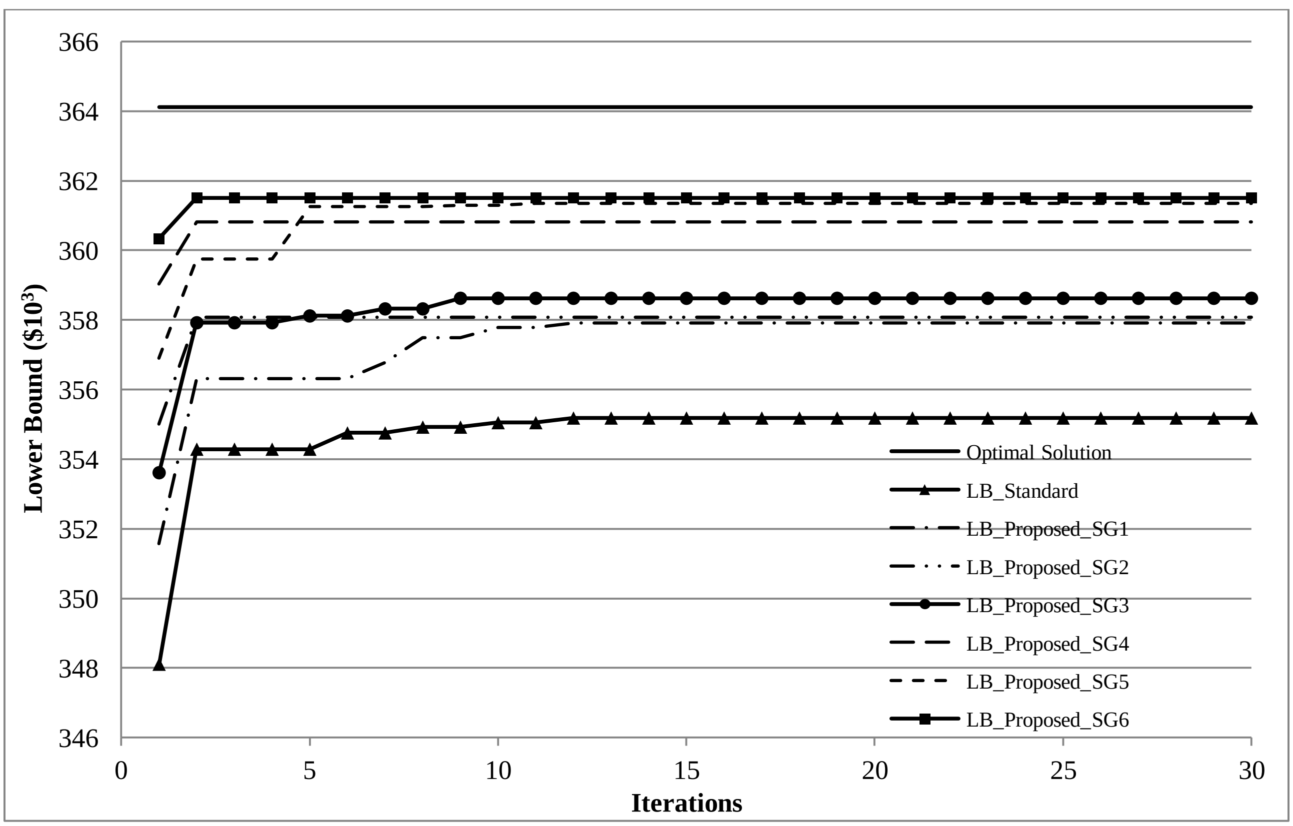

Figure 19: Comparison of the standard vs. proposed approach for 5 process network example

Table 8: Comparison of the standard vs. proposed approach for 5 process network example

	Standard	Proposed SG1	Proposed SG2	Proposed SG3	Proposed SG4	Proposed SG5	Proposed SG6
UB ($\$10^3$)	364.12	364.12	364.12	364.12	364.12	364.12	364.12
LB ($\$10^3$)	355.18	357.92	358.08	358.62	360.82	361.35	361.50
Solution Time (s)	16.32	24.59	15.38	20.40	7.63	12.44	8.9
% Gap	2.52%	1.73%	1.69%	1.53%	<1%	<1%	<1%
# iterations	30	30	30	30	2	5	2

6.2 Oilfield development planning under uncertain field parameters

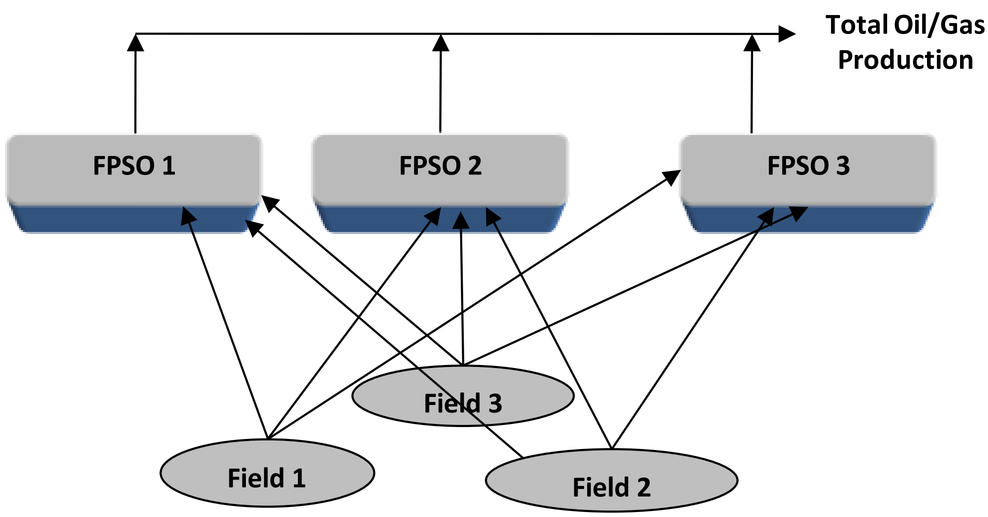

Figure 20: 3 oilfield planning example

Case (i): Uncertainty in the field size only (4 scenarios)

We consider the MILP model by Gupta and Grossmann (2013) for maximizing the expected NPV in the development planning of an offshore oilfield, which is an extension of the previous deterministic model presented in Gupta and Grossmann (2012). The model for all the oilfield planning instances are implemented in GAMS 23.6.3 and run on Intel Core i7, 4GB RAM machine using CPLEX 12.2 solver. In this particular instance, we consider 3 oilfields, 3 potential FPSO's and 9 possible connections among field-FPSO (Figure 20). A total of 30 wells can be drilled in the fields and the planning horizon is 10 years. Field 3 has a recoverable oil volume (field size) of 500 MMbbls. However, there is uncertainty in the size of fields 1 and 2 where each one has two possible realizations (low, high) with equal probability. Therefore, there are a total of 4 scenarios each with a probability of 0.25 (see Table 9). The problem is to determine the investment (FPSO installations and expansions, field-FPSO connections and well drilling) and operating decisions (oil production rate) for this infrastructure with an objective to maximize the total expected NPV (ENPV) over the planning horizon. The optimal ENPV for this problem is \$11.50 x10^9 when the reduced model (MLR) is solved in fullspace, and requires 1184s. Table 10 represents the model statistics for this instance.

Table 9: 3 Oilfield Example (4 Scenarios), case (i)

Scenarios	s1	s2	s3	s4
Field 1 Size (MMbbls)	57	403	57	403
Field 2 Size (MMbbls)	80	80	560	560
Scenario Probability	0.25	0.25	0.25	0.25

Table 10: Model statistics for the 3 Oilfield Example, case (i)

Problem Type	Number of Constraints	Continuous Variables	Discrete Variables	SOS1 Variables
Reduced Model (MLR)	16,473	9,717	876	240
Individual Scenario	3,580	2,390	179	60

Figure 21 compares the performance of the upper bounds obtained at the root node using standard Lagrangean decomposition based on dualizing the initial NACs and removing the conditional NACs (section 4.1.1) with the decomposition approaches proposed in section 5. A termination criteria of either 1% gap or 20 iterations is used. The proposed algorithm based on scenario groups SG1: {(1,2),(3,4)} and SG2: {(1,3),(2,4)} yield stronger upper bounds, \$11.59 x10^9 and \$11.56 x10^9 respectively, than the standard Lagrangean decomposition (section 4.1.1) (\$11.62 x10^9). Additionally, the total computational effort is less with the proposed approach since only 2 subproblems need to be solved at each iteration, and only few iterations are needed to satisfy a 1% of optimality tolerance (Table 11). SG2 performs better than SG1 as can be observed from the total variations in the scenario NPVs with respect to the change in the field sizes as calculated in Table 12 (\$6.63 x10^9 vs. \$4.77 x10^9). This result is similar to the process network example in the previous section. We can also observe that the alternate proposed approach that considers 4 scenario groups (Figure 14(c)) performs well but it is more expensive to solve (429s). It is important to see that the quality of upper bound from SG2 is similar in the first iteration with the quality of UB obtained from the scenario subproblems (section 4.1.1) after 20 iterations (see Figure 21). Moreover, for clarity we only plotted the progress of the upper bounds with iterations and the optimal NPV in Figure 21.

Table 11: Comparison of the various decomposition schemes for oilfield example, case(i)

	Standard	Proposed SG1	Proposed SG2	Proposed Alternate
UB ($\$10^9$)	11.62	11.59	11.56	11.58
LB ($\$10^9$)	11.50	11.50	11.50	11.50
Solution Time (s)	466	382	172	429
% Gap	1.02%	<1%	<1%	<1%
# iterations	20	5	2	3

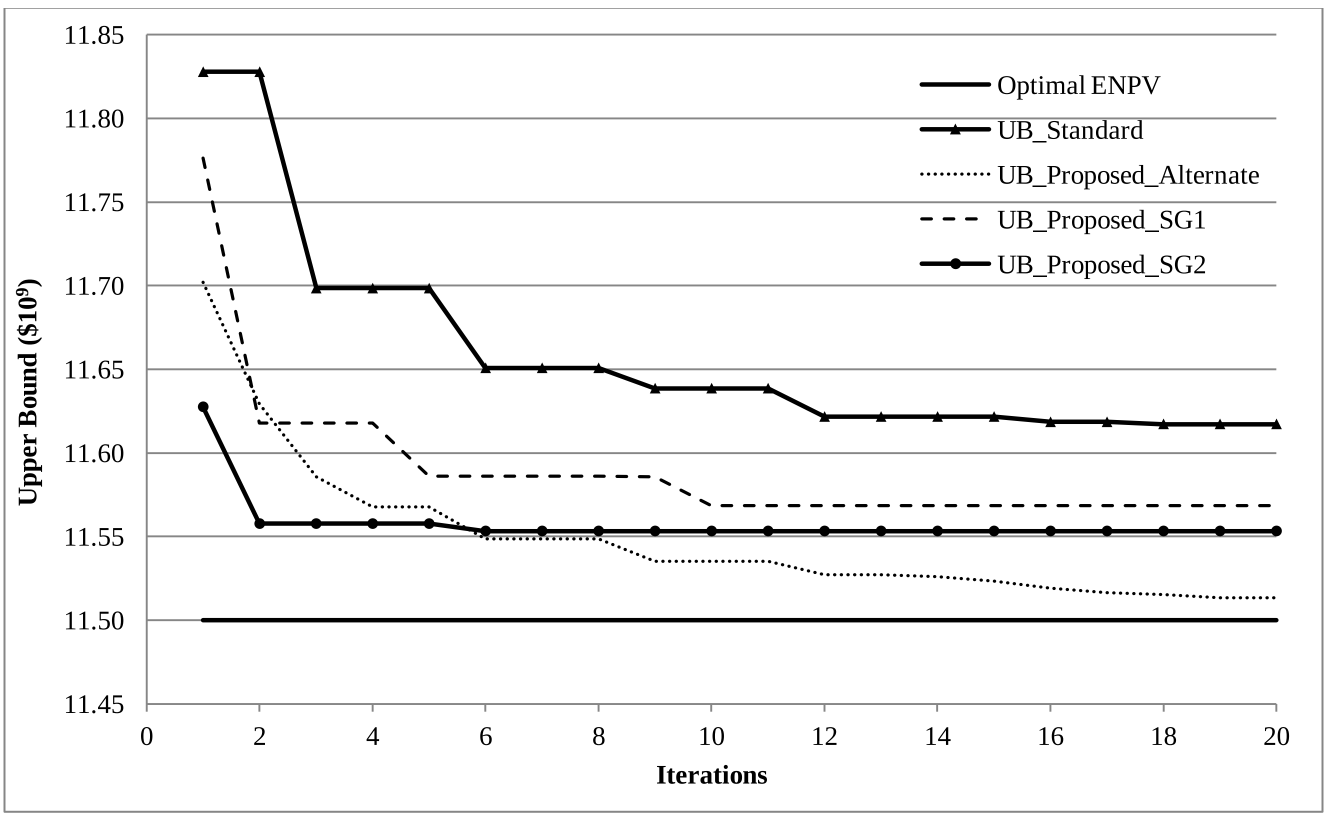

Figure 21: Comparison of the various decomposition schemes for oilfield example 1, case (i)

Table 12: Variations in the objective function value with uncertain parameters, case (i)

(a) Individual Scenario NPV

	NPV ($\$10^9$)
s1	8.95
s2	11.39
s3	12.32
s4	14.65

(b) Scenario groups NPV variations

	SG1	SG2
s1-s2	2.44	-
s3-s4	2.33	-
s1-s3	-	3.37
s2-s4	-	3.26
Total NPV variations ($\$10^9$)	4.77	6.63

Case (ii): Uncertainty in the field size, oil deliverability, WOR and GOR (4 scenarios)

In this case we consider uncertainty in the field size, oil deliverability, water-oil ratio (WOR) and gas-oil-ratio (GOR) for oilfields 1 and 2. Notice that oil deliverability, WOR and GOR are represented by the univariate polynomials in terms of the factional oil recovery as shown in equations (11)-(13) respectively. The uncertainty in these parameters is characterized by the corresponding parameters α_o, α_w and α_g. We assume that the uncertain parameters for a field are correlated and uncertainty in these parameters is resolved at the same time. This allows reducing a large number of scenarios. The two possible combinations of these parameters for each field results in a total of 4 scenarios each with a probability of 0.25 as can be seen in Table 13. The data for the rest of the problem are as in case (i).

$$Q^d = \alpha_o \cdot g(fc) \tag{11}$$

$$wor = \alpha_w \cdot g(fc) \tag{12}$$

$$gor = \alpha_g \cdot g(fc) \tag{13}$$

Figure 22 and Table 14 compare the performance of the upper bounds obtained at the root node using standard Lagrangean decomposition (section 4.1.1) with the proposed decomposition approaches and the similar trends can be observed as in the previous instance. SG2 {(1,3), (2,4)} performs best compared to the other approaches due to the stronger initial bound ($\$12.07\text{x}10^9$). Moreover, since the scenario group set SG2 has a larger total NPV variations ($\$8.70\text{x}10^9$) than set SG1 {(1,2), (3,4)} ($\$5.72\text{x}10^9$), it yields a stronger dual bound. Although, SG1 and the alternate approach are somewhat more expensive compared to the standard decomposition approach, they yield a stronger dual bound in a given amount of solution time. This will eventually reduce the total number of nodes in the branch and bound search tree.

Table 13: 3 Oilfield Example (4 Scenarios), case (ii)

Scenarios		s1	s2	s3	s4
Field 1	Size (MMbbls)	57	403	57	403
	α_o	0.75	1.25	0.75	1.25
	α_w	0.75	1.25	0.75	1.25
	α_g	0.75	1.25	0.75	1.25
Field 2	Size (MMbbls)	80	80	560	560
	α_o	0.75	0.75	1.25	1.25
	α_w	0.75	0.75	1.25	1.25
	α_g	0.75	0.75	1.25	1.25
Scenario Probability		0.25	0.25	0.25	0.25

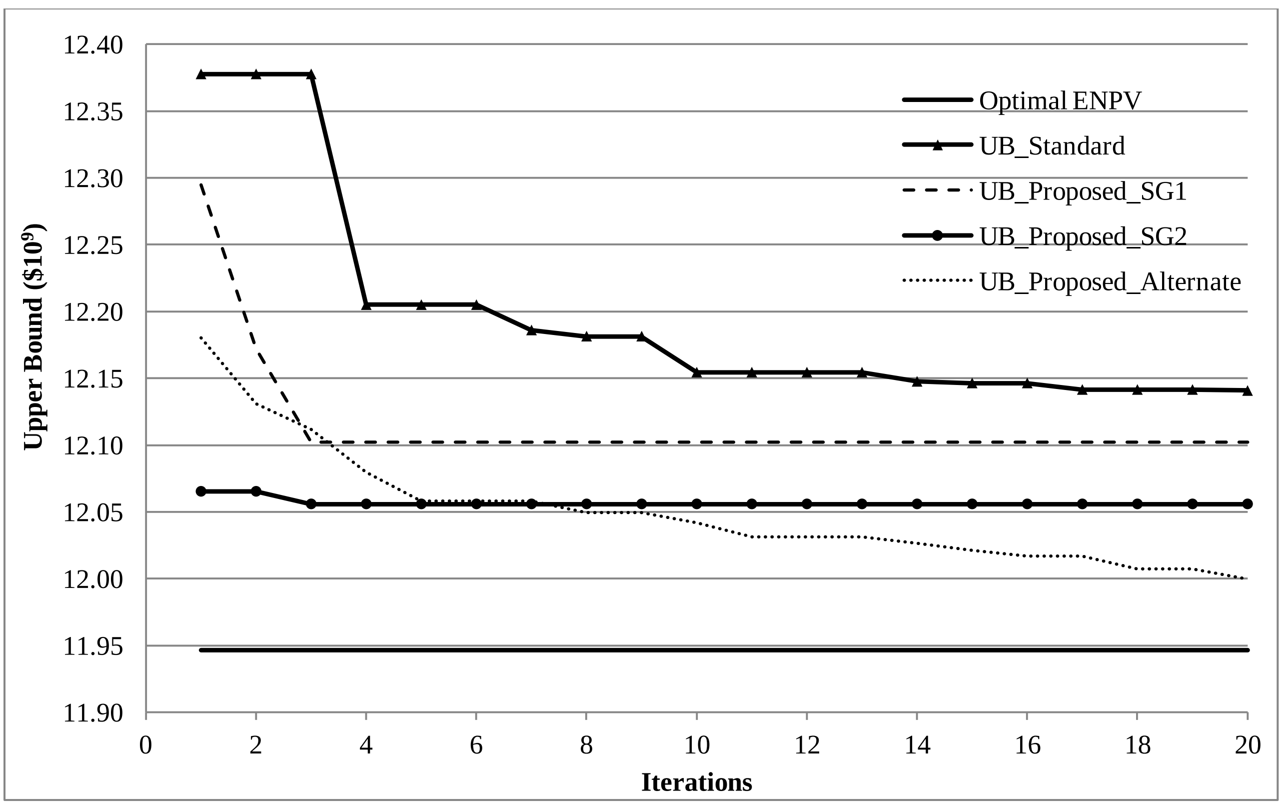

Figure 22: Comparison of the various decomposition schemes for oilfield example, case (ii)

Table 14: Comparison of the various decomposition schemes for oilfield example, case(ii)

	Standard	Proposed SG1	Proposed SG2	Proposed Alternate
UB ($\$10^9$)	12.14	12.10	12.07	12.06
LB ($\$10^9$)	11.94	11.94	11.94	11.94
Solution Time (s)	438	1780	84	1045
% Gap	1.66%	1.28%	<1%	<1%
# iterations	20	20	1	5

Case (iii) and (iv): Extension of the cases (i) and (ii), respectively, for 9 scenarios

In these instances we consider 3 realizations for each uncertain parameter (low, medium, high) compared to two realizations (low, high) in the previous cases (i) and (ii) of oilfield development problem. This results in the corresponding 9 scenario cases (iii) and (iv). Figures 23 and 24 compare the performance of the dual bounds at the root node from various decomposition schemes for these 3 oilfield and 9 scenario instances, whereas Table 15 summarizes the computational results. Since the alternate decomposition (section 5.2.1) is very expensive to solve for these cases, we only compare the proposed approach relying on the scenario groups SG1 {(1,2,3),(4,5,6),(7,8,9)} and SG2 {(1,4,7),(2,5,8),(3,6,9)} with the standard approach (section 4.1.1). We can observe that the initial bound with the proposed strategy ($\$11.93 \text{ x}10^9$) is much better as compared to the final bound obtained from the standard Lagrangean decomposition at the root node ($\$11.96 \text{ x}10^9$) for case (iii). It takes only 2 and 1 iterations in cases (iii) and (iv), respectively, for the proposed approach using set SG2 to reach within 1% of optimality tolerance. On the other hand, the standard and the proposed approach with set SG1 cannot reach within this gap even after 20 iterations or a given time limit of one hour.

Table 15: Comparison of the decomposition schemes for oilfield example, case (iii) and (iv)

	Case (iii)			Case (iv)		
	Standard	Proposed SG1	Proposed SG2	Standard	Proposed SG1	Proposed SG2
UB ($\$10^9$)	11.96	11.92	11.88	12.31	12.26	12.23
LB ($\$10^9$)	11.78	11.78	11.78	12.11	12.11	12.11
Solution Time (s)	1327	>3,600	764	1542	>3,600	439
% Gap	1.47%	1.15%	<1%	1.62%	1.27%	<1%
# iterations	20	10	2	20	8	1

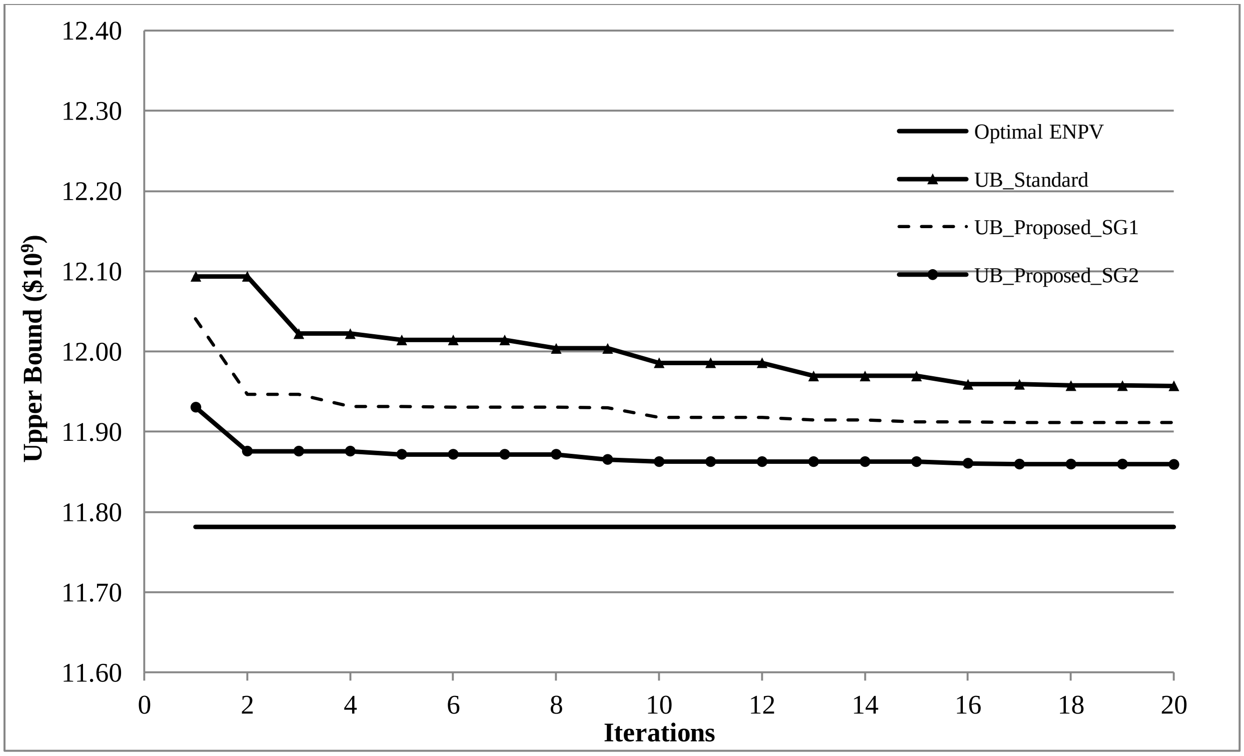

Figure 23: Comparison of the various decomposition schemes for oilfield example, case (iii)

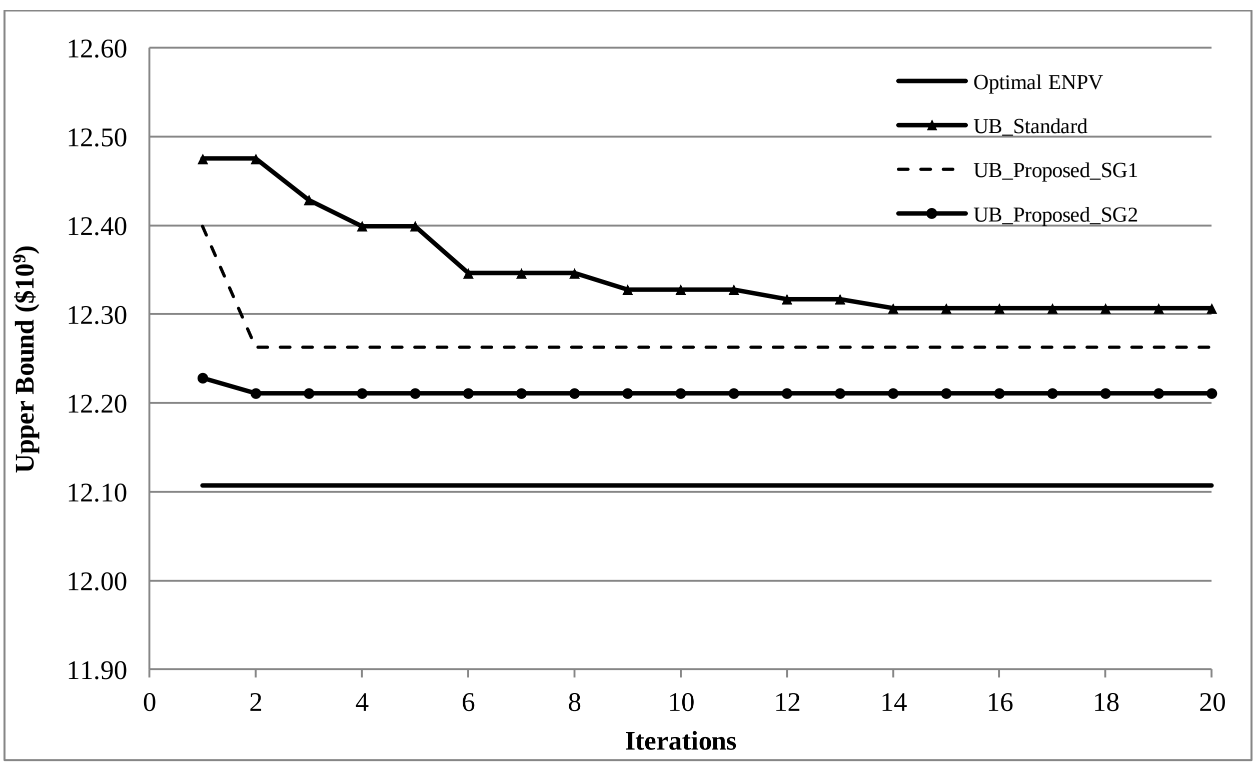

Figure 24: Comparison of the various decomposition schemes for oilfield example, case (iv)

Remarks:

1. Based on the computational results, we can observe that the selection of a particular scenario group set is critical in the proposed approach such as set SG2 performs better than SG1 in all the instances.
2. The increase in the solution time per iteration with the proposed approach is problem specific. For instance, the increase in the solution time per iteration for the process networks examples is not that significant as in the oilfield planning problem. Therefore, if the solution time per iteration for a given problem increases drastically using the proposed decomposition, then one may want to use the standard scenario based approach to explore more nodes quickly in the branch and bound search tree or use subproblems with smaller sizes in the proposed approach.
3. In general, for a given amount of the solution time the proposed approach yields better dual bound and feasible solution as can be seen from the numerical experiments. This is due to the fact that the increase in the solution time per iteration is offset by the significant reduction in the total number of iterations resulting in the lesser total solution time.
4. It should be noted that although the initial gap between lower and upper bounds for the examples presented is not very large for the given data set. However, based on Proposition 1 and computational experiments, we can conclude that the performance of the proposed approach should be similar for the large gap problems given that we select the scenario group sets as described.

7. Conclusions

In this paper, we have proposed a new approach for solving multistage stochastic programs (MSSP) with endogenous uncertainties using Lagrangean decomposition. The proposed approach relies on dividing the fullspace model into scenario groups. Since the number of these scenario groups can be large, there are several alternatives to select a particular set of scenario groups. Therefore, we also presented few rules to identify and formulate a reasonable scenario group set that can be used for the proposed partial decomposition approach within an iterative scheme to update the multipliers. Specifically, the resulting subproblems involve a subset of the NACs as explicit constraints while dualizing and relaxing the rest of these constraints, which enhances the overall performance. An alternate decomposition scheme that may even yield a tighter bound, but usually becomes more expensive for the large cases, is also proposed.

The results on the process network and oilfield planning problems show that the dual bound obtained at the root node from the proposed approaches are stronger than the standard one since the impact of the later time periods is also considered in the subproblems. Moreover, there is a significant reduction in the number of iterations required to converge within a specified tolerance. In most of the cases, even the initial bound with the proposed approach is stronger than the corresponding final bound in the standard approach. Given the tighter bound at the root node, the total number of potential nodes that will be required in the branch and bound search should be smaller and branching rules will be easier to identify. However, the solution time required per iteration in the proposed approach is usually larger as compared to the standard approach, but the difference is problem specific. Therefore, the comparison between the qualities of the bounds obtained within a given amount of solution time should also be considered while selecting a particular decomposition approach for the problems in this class.

References

1. Ahmed, S., Garcia, R., 2003. Dynamic capacity acquisition and assignment under uncertainty. Ann. of Oper. Res. 124, 267–283(17).
2. Ahmed, S., Tawarmalani, M., Sahinidis, N., 2004. A finite branch-and-bound algorithm for two-stage stochastic integer programs. Mathematical Programming, 100(2), 355–377.
3. Birge, J. R., Louveaux, F., 1997. Introduction to stochastic programming. New York, NY: Springer.
4. Boland, N., Dumitrescu, I., Froyland, G., 2008. A multistage stochastic programming approach to open pit mine production scheduling with uncertain geology. http://www.optimizationonline.org/DBHTML/2008/10/2123.html.
5. Caroe, C.C., Schultz, R., 1999: Dual decomposition in stochastic integer programming. Oper. Res. Lett. 24, 37–45.
6. Clay, R. L., Grossmann, I. E., 1997. A disaggregation algorithm for the optimization of stochastic planning models. Computers and Chemical Engineering 21 (7),751-774.
7. Colvin, M., Maravelias, C. T., 2008. A stochastic programming approach for clinical trial planning in new drug development. Computers and Chemical Engineering 32, 2626–2642

8. Colvin, M., Maravelias, C.T., 2010. Modeling methods and a branch and cut algorithm for Pharmaceutical Clinical Trial Planning using Stochastic Programming, European Journal of Operational Research, 203, 205–215.
9. Conejo, A., Castillo, E., Minguez, R., Garcia-Bertrand, R., 2006. Decomposition Techniques in Mathematical Programming. New York: Springer.
10. Fisher, M. L., 1985. An Applications Oriented Guide to Lagrangian Relaxation. Interfaces 15, 10-21.
11. Goel, V., Grossmann, I. E., 2004. A stochastic programming approach to planning of offshore gas field developments under uncertainty in reserves. Computers and Chemical Engineering 28 (8), 1409–1429.
12. Goel, V., Grossmann, I. E., 2006. A class of stochastic programs with decision dependent uncertainty. Mathematical Programming 108 (2-3, Ser. B), 355-394.
13. Goel, V., Grossmann, I. E., El-Bakry, A. S., Mulkay, E. L., 2006. A novel branch and bound algorithm for optimal development of gas fields under uncertainty in reserves. Computers and Chemical Engineering 30, 1076-1092.
14. Guignard, M., 2003. Lagrangean relaxation. Top, 11(2), 151–200.
15. Gupta, V., Grossmann, I. E., 2011. Solution Strategies for Multistage Stochastic Programming with Endogenous Uncertainties. Computers and Chemical Engineering 35, 2235–2247.
16. Gupta, V., Grossmann, I. E., 2012. An Efficient Multiperiod MINLP Model for Optimal planning of Offshore Oil and Gas Field Infrastructure. Industrial and Engineering Chemistry Research 51 (19), 6823–6840.
17. Gupta, V., Grossmann, I. E., 2012. Modeling and Computational Strategies for Offshore Oilfield Development Planning under Complex Fiscal Rules. Industrial and Engineering Chemistry Research 51, 14438−14460.
18. Gupta, V., Grossmann, I. E., 2013. Multistage Stochastic Programming Approach for Offshore Oilfield Infrastructure Planning under Production Sharing Agreements and Endogenous Uncertainties, manuscript in preparation.
19. Held, H., Woodruff, D. L., 2005. Heuristics for multi-stage interdiction of stochastic networks. Journal of Heuristics 11 (5-6), 483-500.

20. Ierapetritou, M. G., Pistikopoulos, E. N., 1994. A novel optimization approach of stochastic planning models. Industrial and Engineering Chemistry Research 33, 1930-1942.
21. Jonsbraten, T., 1998. Optimization models for petroleum field exploitation. PhD thesis, Norwegian School of Economics and Business Administration.
22. Li, Z., Ierapetritou, M., 2012. Capacity expansion planning through augmented Lagrangian optimization and scenario decomposition. AIChE Journal, 58(3), 871–883.
23. Mouret, S., Grossmann, I. E., Pestiaux, P., 2011. A new Lagrangian decomposition approach applied to the integration of refinery planning and crude-oil scheduling. Computers and Chemical Engineering 35, 2750-2766
24. Oliveira, F., Gupta, V., Hamacher, S., Grossmann, I. E., 2013. A Lagrangean Decomposition Approach for Oil Supply Chain Investment Planning under Uncertainty with Risk Considerations. Computers and Chemical Engineering 50, 184–195.
25. Ruszczynski, A., 1997. Decomposition methods in stochastic programming. Math. Programming (Ser. B) 79, 333-353.
26. Sahinidis, N.V., 2004. Optimization under uncertainty: State-of-the-art and opportunities. Computers and Chemical Engineering 28 (6–7), 971–983.
27. Schultz, R., 2003. Stochastic programming with integer variables. Mathematical Programming 97 (1–2), 285–309.
28. Solak, S., 2007. Efficient solution procedures for multistage stochastic formulations of two problem classes. PhD thesis, Georgia Institute of Technology.
29. Tarhan, B., Grossmann, I. E., 2008. A multistage stochastic programming approach with strategies for uncertainty reduction in the synthesis of process networks with uncertain yields. Computers and Chemical Engineering 32, 766-788.
30. Tarhan, B., Grossmann, I. E., Goel, V., 2009. Stochastic programming approach for the planning of offshore oil or gas field infrastructure under decision-dependent uncertainty. Industrial and Engineering Chemistry Research 48 (6), 3078-3097.
31. Tarhan, B., Grossmann, I. E., Goel, V., 2011. Computational strategies for non-convex multistage MINLP models with decision-dependent uncertainty and gradual uncertainty resolution. Annals of Operations Research, http://dx.doi.org/10.1007/s10479-011-0855-x.

32. Tarhan, B., Gupta V., Grossmann, I. E., 2013. Improving Dual Bound for Stochastic MILP Models using Sensitivity Analysis, manuscript in preparation.
33. Viswanath, K., Peeta, S., Salman, S. F., 2004. Investing in the links of a stochastic network to minimize expected shortest path length. Tech. rep., Purdue University.