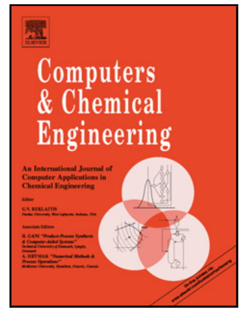

Title: Symmetry Breaking in MILP Formulations for Unit Commitment Problems

Author: Ricardo M. Lima Augusto Q. Novais

PII: S0098-1354(15)00362-2
DOI: http://dx.doi.org/doi:10.1016/j.compchemeng.2015.11.004
Reference: CACE 5317

To appear in: *Computers and Chemical Engineering*

Received date: 5-4-2015
Revised date: 12-11-2015
Accepted date: 17-11-2015

Highlights
Symmetry Breaking in MILP Formulations for Unit Commitment Problems”
by Ricardo M. Lima and Augusto Q. Novais
Submitted to *Computers & Chemical Engineering*.

- Symmetry issues are discussed in the context of UC formulations.
- Three sets of SBC are proposed and analyzed.
- Three well-known UC MILP models are used to implement the SBC.
- The results presented are completely disruptive compared with the results presented in the literature.
- The SBC derived have a clear advantage over general methods available in two MILP solvers.

Symmetry Breaking in MILP Formulations for Unit Commitment Problems

Ricardo M. Lima[a,*], Augusto Q. Novais[b]

[a] *Computer, Electrical and Mathematical Sciences & Engineering Division, King Abdullah University of Science and Technology, (KAUST), Thuwal 23955-6900, Kingdom of Saudi Arabia*
[b] *CEG-IST, Instituto Superior Técnico, Universidade de Lisboa, Av. Rovisco Pais, 1049-001 Lisboa, Portugal*

Abstract

This paper addresses the study of symmetry in Unit Commitment (UC) problems solved by Mixed Integer Linear Programming (MILP) formulations, and using Linear Programming based Branch & Bound MILP solvers. We propose three sets of symmetry breaking constraints for UC MILP formulations exhibiting symmetry, and its impact on three UC MILP models are studied. The case studies involve the solution of 24 instances by three widely used models in the literature, with and without symmetry breaking constraints. The results show that problems that could not be solved to optimality within hours can be solved with a relatively small computational burden if the symmetry breaking constraints are assumed. The proposed symmetry breaking constraints are also compared with the symmetry breaking methods included in two MILP solvers, and the symmetry breaking constraints derived in this work have a distinct advantage over the methods in the MILP solvers.

Keywords: Scheduling, Symmetry Breaking, MILP, Unit Commitment

1. Introduction

The thermal Unit Commitment (UC) problem has been the subject of multiple works and tackled employing different formulations, solution strategies and solvers, see Padhy (2004) for a review. The UC problem has been mainly studied in the Power Systems and Operations

*Corresponding author.
Email addresses: ricardo.lima@kaust.edu.sa (Ricardo M. Lima), augusto.novais@lneg.pt (Augusto Q. Novais)

Research communities, but recently the Process Systems Engineering (PSE) community has put some efforts in the efficient solution of UC problems, and has used parts of UC models to describe the operations of some units. For example, Marcovecchio et al. (2011, 2014) proposed a branch and cut search method tailored to solve deterministic UC problems, while Zondervan et al. (2010) compared the performance of different solvers to solve nonlinear versions of UC problems. In the context of planning models for energy generation expansion, Siirola and Watson (2012) solved UC problems using a modeling and optimization framework based on Python, and in the same context Koltsaklis and Georgiadis (2015) solved UC problems to provide detailed information to a planning model. Other examples where the UC problems are solved or discussed by PSE community can be found in Friedman et al. (2013); Hodge et al. (2011); Liu et al. (2013); Soroush and Chmielewski (2013); Xiao et al. (2011). Some of the modeling features of UC problems have been discussed and used by the PSE community to model the operations of specific units (Mitra, 2013). For example, some equations that are typical in UC models have been used to model the operations of boilers that feed steam to combined heat and power plants (Mitra et al., 2013), and to model specific operations of air separation units (Mitra et al., 2012). This interest in the UC problem is motivated by its relevance in terms of economic and environmental impacts, due to the research challenges it poses when large-scale systems are considered, but also because of the scheduling nature of UC problems that falls into the interests of the PSE community.

The traditional UC problem can be recast as a scheduling problem of a set of thermal generators that generate electricity under specific constraints. The objective is to minimize the generation cost to meet the energy demand in each period of time, for a time horizon of 24 hours. Currently, UC problems are incorporated into more comprehensive models that may include network transmission constraints, market interactions, the presence of uncertainty, and renewable generators. However, as the level of complexity of these models increases, their applicability may require the utilization of decomposition strategies, which will depend on the efficient solution of UC models at some stage. For example, Hedman et al. (2010) adopted a decomposition method to address the co-optimization of the generation and topology of a network, where the generation is represented by a UC problem that is solved isolated. Similarly, Zhou et al. (2012) extended a deterministic UC formulation to a stochastic version in order to study the impact of wind forecasts on the determination of optimal reserves. In both cases, the solution framework clearly depends on the efficient solution of a UC problem.

UC problems have been solved using different techniques, namely meta-heuristics, priority lists, sequential UC, Dynamic Programming (DP), Lagrangian Relaxation (LR), and MILP solvers, see Padhy (2004) for a review. Each approach has well known strong and weak points. DP and LR are very popular in industry (Dereu and Grellier, 2010; Frangioni, 2010; Hechme-Doukopoulos et al., 2010) and they are widely disseminated in the literature (Fan et al., 2002; Frangioni et al., 2008; Li and Shahidehpour, 2005; Redondo and Conejo, 1999). LR relies on a strong mathematical foundation, its capacity to decompose large problems into smaller subproblems and by the quality of the calculated bounds. However, LR requires specific strategies that may depend on the problem, which reduces its flexibility.

MILP techniques offer excellent modeling capabilities and flexibility, and in recent years, they have been increasingly studied to address UC problems (Arroyo and Conejo, 2000; Carrion and Arroyo, 2006; Chang et al., 2001; Frangioni et al., 2009; Hedman et al., 2010; Ostrowski et al., 2012; Sioshansi et al., 2008). Furthermore, they have also been adopted by Independent System Operators (ISO), as reported by Hedman et al. (2010) and Sioshansi et al. (2008). Their increasing applicability has been leveraged by significant advancements in two areas that had limited their application in the past, namely computational power and MILP solvers. These two fields have been evolving due to the availability of faster CPU processors with multiple cores and the increasing performance of MILP solvers. The development of the MILP solvers has been supported by the implementation of new cuts based on polyhedral theory, as well as the utilization of heuristics and meta-heuristics within the B&B search. The combination of these features help finding integer solutions while keeping the logic inherent to the B&B algorithm (Bixby and Rothberg, 2007; Lima and Grossmann, 2011; Rothberg, 2007). In addition, the efficient application of MILP solvers depends on three model features: 1) a good linear relaxation; 2) the size of the formulation;

and 3) the presence of symmetric solutions in the formulation.

Deriving good linear relaxations for UC models has been the subject of several works. Ostrowski et al. (2012) proposed a tight MILP formulation that is based on the addition of valid inequalities that are facets of the ramp up and down constraints associated with each generator. These inequalities, though in a large number, have a considerable impact on the tightness of the linear relaxation. The combination of a tight linear relaxation with MILP solvers containing useful node heuristics at the root node allows their formulation to reach solutions, early in the B&B tree search, with relatively small optimality gaps. The formulation proposed by Ostrowski et al. (2012) is built on previous work from Arroyo and Conejo (2000); Carrion and Arroyo (2006) and Rajan and Takriti (2005). The two formulations developed in Arroyo and Conejo (2000) and Carrion and Arroyo (2006) feature minimum up and down times, involved startup costs, power ramp down and up constraints, demand satisfaction and spinning reserves. The formulation from Carrion and Arroyo (2006) features a more compact formulation than Arroyo and Conejo (2000). The former uses one set of binary variables that indicates the on/off state of the units in each time period while Arroyo and Conejo (2000) used a formulation with three sets of binary variables. The three sets of binary variables represent the on/off state, the startup, and the shutdown of the units. The utilization of either one or three sets of binary variables has been discussed extensively in the literature. Interestingly, different authors have arrived at different conclusions about the relative value of each formulation (Carrion and Arroyo, 2006; Hedman et al., 2010; Ostrowski et al., 2012). Rajan and Takriti (2005) and Lee et al. (2004) have proposed convex-hull formulations for the minimum up and down constraints that lead to tight linear relaxations. The former developed a relaxation based on three sets of binary variables while the latter built the relaxation based only on the set of binary variables that indicates the on/off state of the units. Another contribution has dealt with valid inequalities for the linearization of the quadratic objective function (Frangioni et al., 2009).

The works mentioned above have focused on deriving tight and compact MILP formulations that, when combined with MILP solvers with good heuristics may lead to relatively fast integer solutions with small optimality gaps. While it is ubiquitous that a solution with a small optimality gap may be acceptable for some real world problems, in some other situations suboptimal solutions may have a significant economic impact, as reported by Sioshansi et al. (2008).

1.1. Symmetry in Unit Commitment problems

The existence of symmetry in UC problems has been reported in the literature in two areas: 1) benchmark problems (Morales-Espana et al., 2013; Ostrowski et al., 2012); and 2) real world generation systems (Alemany et al., 2014; Dekrajangpetch et al., 1999; Dentcheva et al., 1997).

A careful analysis of the results obtained with the typical sets of benchmark problems employed in the literature, suggested us that the difficulty in closing the optimality gap might be due to the level of symmetry introduced by those problems. These benchmarks are usually based on the data set from Kazarlis et al. (1996), which contains several sets of indistinguishable generators. The term indistinguishable is adopted from Sherali and Smith (2001) to classify objects with equal features that just differ on the index assigned. There is a significant number of works in the literature that use benchmark problems involving sets of indistinguishable generators to assess the quality of MILP formulations (Carrion and Arroyo, 2006; Morales-Espana et al., 2013; Ostrowski et al., 2012).

The presence of symmetry in MILP formulations leads to multiple solutions with equal objective function values that force MILP solvers to explore a vast numbers of nodes of the tree. However, in spite of the impact of symmetric solutions and the type of benchmark problems used, symmetry breaking techniques have been almost neglected in UC MILP formulations.

The effects of indistinguishable generation units in UC problems has been studied before in the context of LR (Alemany et al., 2014; Benhamida and Abdelbar, 2010; Dekrajangpetch et al., 1999; Dentcheva et al., 1997; Zhai et al., 2002). In LR the symmetry was identified as a source of convergence problems, leading to suboptimal solutions or even infeasible solutions Dekrajangpetch et al. (1999). Therefore, LR has also required specific solution approaches to deal with symmetry.

One of the first approaches that is in general used to break the symmetry relies on adding a slight random perturbation in the cost factors in the objective function. However, these perturbations may have an adverse impact on the performance of the solution algorithms (Margot, 2010), whereby more elegant approaches may have a superior performance (Margot, 2010).

Symmetry breaking techniques can be broadly divided into (Liberti, 2012): specific and general purpose. Specific techniques rely on two different approaches; the first involves the definition of additional constraints tailored to the particular problem to solve, and the second is based on branching rules in the B&B search.

Examples of the application of specific techniques are typically found in scheduling problems. In these problems two cases may arise: 1) processes with machines with identical characteristics (Sherali, 2001); or 2) problems where the assignment of n products has to be made to a number of m slots, with $m > n$ (Lima et al., 2011). Recently Symmetry Breaking Constraints (SBC) are discussed in the context of a real world UC problem in Alemany et al. (2014). The SBC implemented in that work are further discussed in Section 4.1. Lima et al. (2013) proposed specific SBC for the short term scheduling of hydro plants where multiple sets of turbines with identical features are present in the same hydro plant. The SBC implemented are used within the MILP formulations, and they are further used within a spatial branch and bound (B&B) algorithm to tighten the bounds on the variables involved in the convex envelopes built for the bilinear terms. The results showed that SBC have a substantial impact on the performance of the MILP problems within a global optimization framework, as well as within local MINLP solvers that rely on solving MILP problems.

General purpose techniques are based on the automatic identification of symmetry and generation of symmetry breaking strategies. Ostrowski et al. (2014) reported the application of a general purpose symmetry breaking technique based on a modified orbital branching algorithm to a thermal UC MILP model. Their results show that modified orbital branching performs better than orbital branching and the B&B method of CPLEX, in terms of computational time to meet the optimality gap of 0.1%. Those authors assess their method on the same set of benchmark problems used in this work. Alemany et al. (2014) tested general SBC implemented in an off-the-shelf MILP solver against specific cuts implemented by the authors, and the former revealed to be less efficient.

In this work, a set of SBC is proposed that improve the performance of MILP solvers in terms of computational time and number of nodes explored during the B&B search. These constraints are motivated by the presence of multiple sets of indistinguishable generators in the benchmark problems used in the literature. The SBC are implemented in three UC formulations available in the literature, and their effectiveness demonstrated in closing the optimality gap for a class of problems that defy solution or require extensive CPU times.

The proposed SBC are a clear and easy way to break the symmetry in the models and the instances solved. However, to the best of our knowledge these SBC were not presented before for UC problems. We are careful enough to discuss the limitations of the proposed constraints, and analyze constraints that may eliminate the optimal solution from the models studied in this work. Therefore, the SBC are formulated taking into consideration that the models used involve features such as minimum up and down times and ramp rates constraints, which pose some restrictions on the derivation of symmetry breaking constraints.

The computational results presented here are completely disruptive compared with the results presented by Ostrowski et al. (2012). Those authors could not close the optimality gap of several problems in thousands of seconds while the same problems are solved to optimality 0.0% in hundreds of seconds in this work. Therefore, the scientific contribution results from the derivation of the proper symmetry breaking constraints and new computational results that to the best of our knowledge were not published before.

The structure of the paper is the following: In Section 2, the base MILP models are referenced. Section 3 shows some motivating examples showing distinct types of symmetries and their impact on the performance of the MILP solvers. Section 4 describes the proposed SBC. Section 5 presents the computational results, and the final conclusions are drawn in Section 6.

2. MILP models

A general formulation of a UC model is given by Wood and Wollenberg (1996):

$$\min cost = \sum_i \sum_t (cp_{i,t} + cu_{i,t} + cd_{i,t}), \tag{1}$$

$$\sum_i p_{i,t} = D_t, \quad \forall t, \tag{2}$$

$$\sum_i \overline{p}_{i,t} \geq D_t + SR_t, \quad \forall t, \tag{3}$$

$$cp_{i,t}, cu_{i,t}, cd_{i,t}, \overline{p}_{i,t}, p_{i,t} \in \Pi_{i,t}, \quad \forall i, t, \tag{4}$$

where the objective is to minimize the total operating cost encompassing the production costs, $cp_{i,t}$, the startup costs, $cu_{i,t}$, and the shutdown costs, $cd_{i,t}$, subject to the power balance constraint (2), the spinning reserves constraint (3), and the feasible region of operation of the generators, $\Pi_{i,t}$, constraint (4).

Several seminal works (Arroyo and Conejo, 2000; Carrion and Arroyo, 2006; Frangioni et al., 2008, 2009; Morales-Espana et al., 2013; Ostrowski et al., 2012), have proposed UC MILP models improvements without considering transmission constraints. These UC MILP models can be of practical interest when decomposition methods are used for more comprehensive problems, where the UC problem is isolated and solved without the transmission constraints.

In this work, three alternative UC models are taken from the literature:

1. Model I from Ostrowski et al. (2012);
2. Model II from Ostrowski et al. (2012);
3. Model III from Morales-Espana et al. (2013).

These models are used without modifications to describe the feasible region of operation. The three models considered include features such as minimum up and down time and ramp up and down constraints that prevent the utilization of simple symmetry breaking constraints. The models are equivalent in terms of the optimal value of the objective function but have distinct characteristics in terms of size, quality of the continuous linear relaxation, and computational performance. See Appendix A for a description of the base models.

These three models were chosen because they encompass some of the best formulations in terms of the tightness and compact representation of UC MILP models, as it is discussed in detail in Ostrowski et al. (2012), and Morales-Espana et al. (2013). However, in these two references, the authors did not propose any symmetry breaking constraints in their models. Furthermore, they advocate that problems with indistinguishable generators give rise to more severe test problems (Morales-Espana et al., 2013; Ostrowski et al., 2012). In general, there is a trend to assess the impact of the size and tightness of UC MILP formulations on the computational performance using problems that exhibit significant symmetry. However, this may not be appropriate to evaluate the performance of MILP formulations since the symmetry introduces noise on the ability of the MILP solvers to close the optimality gap. Therefore, in this work relevant aspects related to symmetry breaking that were not considered by two of the most recent UC MILP models proposed in the literature are discussed.

3. Motivating examples

UC MILP formulations are usually tested on benchmark problems that are built over a base set of unique thermal generators, which by cloning are then employed to generate problems of larger dimensions. This scale-up method originates problem instances of different sizes with multiple sets of generators with identical characteristics and initial state. However, this approach to generate instances has been widely used in the literature (Carrion and Arroyo, 2006; Marcovecchio et al., 2014; Morales-Espana et al., 2013; Ostrowski et al., 2012). In this section, particular instances of the two case studies analyzed in Section 5 are used to illustrate some typical results obtained in terms of computational performance and commitment of units with this type of benchmark problems.

The first example uses an instance of Case 2 with 35 generators, see Table 1. The results presented in Table 1 show that the MILP solver CPLEX cannot get the gap closed after 10 hours of CPU time. At the end of the root node, the relative optimality gaps obtained with the three models used are 0.10%, 0.09%, and 0.09%, while after one hour are reduced to 0.01% for Models I and II, and 0.04% for Model III. However and despite these small gaps, the solver cannot close them within an additional nine hours. These results suggest that the linear relaxations are considerably tight, and also that the current MILP solver can find very good integer solutions early in the B&B tree. However, due to the solutions symmetry, the B&B algorithm has to explore a large number of nodes before closing the optimality gap. These computational findings motivated the exploration of SBC as a means to accelerate the optimality gap closure. This subject has already been identified by Ostrowski et al. (2012), and Alemany et al. (2014), but not fully explored in UC MILP formulations. The next two examples identify some cases of symmetry in the solutions of the UC problems. The first example is based on the results obtained with the instance of 30 generators drawn from Case 1, described later. This problem has three sets of generators with identical characteristics and initial state, namely 1-10, 11-20, and 21-30. For this instance, the matrices presented in (5) and (6) correspond to two optimal solutions. These matrices show the values of $u_{i,t}$, which represent the generation schedules in terms of the on/off states during the time horizon. Each row represents a generator and each column a period of one hour, with only committed generators being shown. The symmetry in these solutions is easily observed since the permutation of specific rows within one matrix leads to the other. For example, if we apply the following operations in the first matrix, the second matrix is replicated: 1) permutation of two rows of generators that are both committed, e.g. permuting row 1 with row 6 in the first matrix; and 2) permutation of one row of one generator that is

Table 1: Case 2, instance with 35 generators. Partial and final results.

	End of RN		T=3,600s	T=36,000s		
Model	G (%)	T (s)	G (%)	G (%)	NEvl	NOpn
I	0.10	28	0.01	0.004	14,961,164	5,990,709
II	0.09	29	0.01	0.001	17,544,417	405,897
III	0.09	5	0.04	0.035	1,277,703	5,013,922

RN - Root node, G - Relative optimality gap, T - CPU time,
NEvl - Number of nodes evaluated, NOpn - Number of nodes open.

committed with the row of other generator that is not committed during the time horizon, e.g. permuting generators 17 and 14, and generators 27 and 25 in the first matrix. Any of these permutations transforms matrix (5) into (6) without changing the value of the objective function.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1									1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
4	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
5	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
6	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
7	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
8	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
9	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
10	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
17	1	1	1	1	1	1	1	1	1	1	1	1	1	1										
27																	1	1	1	1	1			

(5)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
4	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
5	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
6									1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
7	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
8	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
9	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
10	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
14	1	1	1	1	1	1	1	1	1	1	1	1	1	1										
25																	1	1	1	1	1			

(6)

4. Symmetry breaking constraints

In this work, three different sets of SBC constraints are investigated, and their value is assessed by relative comparison, and by comparison with the base models without the SBC. The first approach implemented to break the symmetry is based on the identification of a new state that indicates if the generator is committed at any time during the time horizon. This logical proposition can be represented by the disjunction

$$\begin{bmatrix} YON_i \\ \sum_t u_{i,t} \geq 1 \end{bmatrix} \vee \begin{bmatrix} \neg YON_i \\ \sum_t u_{i,t} \leq 0, \end{bmatrix}, \quad \forall i, \tag{7}$$

where YON_i denotes a boolean variable that is true if at least one $u_{i,t} = 1$, and false if the generator is not committed during the time horizon. The above disjunction can be rewritten using a big-M or a convex-hull reformulation (Balas, 1985). Note that both reformulations lead to the same representation, see Vecchietti et al. (2003). Therefore, (7) can be represented as

$$\sum_t u_{i,t} \geq yon_i, \quad \forall i, \tag{8}$$

$$\sum_t u_{i,t} \leq |T| yon_i, \quad \forall i, \tag{9}$$

where yon_i is a binary variable for each generator i, whereby $yon_i = 1$ if generator i is committed during the time horizon and is $yon_i = 0$ otherwise, and $|T|$ is the total number of time periods. Equation (8) ensures that for $yon_i = 1$ then at least one $u_{i,t} = 1$, while (9) forces yon_i to be nonzero, i.e. $yon_i = 1$, if at least there is one $u_{i,t} \geq 1$. Equation (9) resembles a well-known (Wolsey, 1998) constraint in integer programming that can also be written as:

$$u_{i,t} \leq yon_i, \quad \forall i, t. \tag{10}$$

With the above equations, the following symmetry breaking constraint is added to the formulation:

$$yon_i \geq yon_{i+1}, \quad \forall i, i+1 \in M_i. \tag{11}$$

(11) forces generator i to be committed if generator $i+1$ is committed. Note that (11) is only valid for the set of generators with identical characteristics, M_i, and is independent of the time and duration of the commitment of the units.

In order to break the symmetry posed by the permutation of generators that are both committed, the following constraint is proposed:

$$\sum_t u_{i,t} \geq \sum_t u_{i+1,t}, \quad \forall i, i+1 \in M_i. \tag{12}$$

This equation sorts the solution matrix u, forcing the generators with most periods committed having the lowest index. Equation (12) is similar to the equations presented in Margot

(2010) and Rey (2004). Both Equations (11) and (12) force a hierarchical structure of commitment, and their derivation may suggest that they are equivalent, but they are not. The first set of three equations, (8), (9), and (11) are related to the existence or not of commitment during the time horizon. These equations do not force an order based on the number of periods committed by each generator. While (12) forces generator i to be committed a number of periods greater or equal than generator $i+1$, $\forall i, i+1 \in M_i$, which establishes an hierarchical decision order over the total extent of the commitment of the generators. Based on these equations, three groups of constraints are considered to break the symmetry, see Table 2.

Table 2: Groups of equations to break symmetry.

Reference	Equations
S1	(12)
S2	(8), (9), (11), (12)
S3	(8), (10), (11), (12)

From a polyhedral perspective, considering only constraints (12) is equivalent to the sets S2 and S3. Any solution that satisfies constraints (12) also satisfies constraints (8), (9), (10), and (11). However, it is non-trivial to infer which combination of UC MILP model and SBC group will have the better performance with a given MILP solver. This is justified by the combination of different model features such as the size and the tightness of the continuous linear relaxation, which combined with the algorithmic strategies of the solvers may lead to similar formulations to have distinct computational results.

4.1. Remarks

The adopted SBC in this work are useful for problems with sets of indistinguishable generators, which are assumed to be entirely independent of each other. In addition, they will not break all possible symmetries in the models. For example, they are not efficient when two solutions have sub-sets of indistinguishable generators committed during the same total number of time periods, as in

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1											1	1	1	1										
2																			1	1	1	1		

(13)

and in the specific case described next. As an illustrative example, consider the matrices (14) to (16) that show three UC commitment solutions of the same problem, with the same objective function value. These commitments respect the proposed symmetry breaking constraints, but they are still symmetric. The source of symmetry arises from the assignment in $t = 23$, which may be assigned to generators 1, 2, or 3, without changing the objective function value.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1												1	1	1			1	1	1	1	1	1	1	
2												1	1	1					1	1	1	1		
3												1	1	1							1	1		

(14)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1												1	1	1			1	1	1	1	1	1		
2												1	1	1					1	1	1	1	1	
3												1	1	1							1	1		

(15)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1												1	1	1			1	1	1	1	1	1		
2												1	1	1					1	1	1	1		
3												1	1	1							1	1	1	

(16)

Therefore, the models that would be suitable for the proposed SBC may still exhibit some forms of symmetry that limit the applicability of MILP solvers. However, it is hard to derive general approaches that will break the symmetry presented above and still be globally valid.

Recently, Alemany et al. (2014) proposed the following equation to break the symmetry between indistinguishable generators:

$$u_{i,t} \geq u_{i't}, \quad \forall i \neq i' \in M_i, t. \tag{17}$$

This equation is a strong cut to break the symmetry between generators, but it is only valid under specific conditions. It may not be valid, or it may cut-off the optimal solution, whenever the generators and the UC solution have particular characteristics. For example, if the generators have constraints such as minimum up and down times, ramp rate limits, and $\sum_i p_{i,t} = D_t,\ \forall t$, and the optimal solution is similar to the matrix (18), then (17) cuts off the optimal solution. Therefore, equation (17) is not of general applicability, and it is not considered in this work, where the focus is only on equations that do not eliminate the optimal solution.

$$\begin{array}{c|cccccccccccccccccccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 19 & 20 & 21 & 22 & 23 & 24 \\ \hline 1 & & & & & & & & & & & & & & & & & 1 & 1 & 1 & 1 & & & & \\ 2 & & & & & & & & & & & & & & & & & & & & 1 & 1 & 1 & 1 & \end{array} \tag{18}$$

5. Computational results

The primary goal of the computational results is to demonstrate the impact that symmetry has on the computational efficiency of UC MILP formulations. With this aim, two versions of the three base Models I, II, and III are applied to solve several instances: a base version and a modified one with the incorporation of the SBC. The performance of the models with the SBC is also compared with the symmetry breaking methods implemented in two MILP solvers.

In this work, the stopping criterion is set on a maximum CPU time of 7,200s, and then the number of nodes, the optimality gap, and the CPU time are used to assess the efficiency of the models. The MILP models are implemented in the modeling system GAMS and solved with the MILP solvers CPLEX 12.6.0, and GUROBI 5.6.0, in a computer with an Intel Core i7@3.07GHz, 64 bits CPU, and 8Gb of RAM.

The case studies addressed in this work are based on the benchmark data proposed by Kazarlis et al. (1996) and widely used by several authors (Carrion and Arroyo, 2006; Morales-Espana et al., 2013; Ostrowski et al., 2012). Two case studies are considered:

1. Case 1 - This case involves nine instances. Table 3 provides the distribution of generators by instance. The spinning reserves are specified as 10% of the demand.
2. Case 2 - This case considers 15 instances based on Ostrowski et al. (2012). Table 3 provides the distribution of generators by instance. The spinning reserves are set to 3% of the demand.

The differences between Case 1 and Case 2 rely on the distribution of generators, demand data, and spinning reserves specifications. The data for the generators and demand are

Table 3: Distribution of generators by instance. The references of the instances are in the column with the headers C1 and C2.

	Type of generator										
C1	1	2	3	4	5	6	7	8	9	10	TG
1	10	10	0	0	0	0	0	0	0	0	20
2	10	10	10	0	0	0	0	0	0	0	30
3	10	10	10	10	0	0	0	0	0	0	40
4	10	10	10	10	10	0	0	0	0	0	50
5	10	10	10	10	10	10	0	0	0	0	60
6	10	10	10	10	10	10	10	0	0	0	70
7	10	10	10	10	10	10	10	10	0	0	80
8	10	10	10	10	10	10	10	10	10	0	90
9	10	10	10	10	10	10	10	10	10	10	100

	Type of generator										
C2	1	2	3	4	5	6	7	8	9	10	TG
1	12	11	0	0	1	4	0	0	0	0	28
2	13	15	2	0	4	0	0	1	0	0	35
3	15	13	2	6	3	1	1	3	0	0	44
4	15	11	0	1	4	5	6	3	0	0	45
5	15	13	3	7	5	3	2	1	0	0	49
6	10	10	2	5	7	5	6	5	0	0	50
7	17	16	1	3	1	7	2	4	0	0	51
8	17	10	6	5	2	1	3	7	0	0	51
9	12	17	4	7	5	2	0	5	0	0	52
10	13	12	5	7	2	5	4	6	0	0	54
11	15	13	8	7	12	7	2	1	0	0	65
12	20	15	10	8	6	5	6	5	0	0	75
13	25	21	12	3	1	7	7	9	0	0	85
14	30	26	15	10	2	6	3	3	0	0	95
15	35	30	0	15	5	6	1	8	0	0	100

C1 - Case 1; C2 - Case 2; TG - Total number of generators

available in Appendix B.

The results are divided into two sections. In the first, 24 instances are used to assess the relative merit of the three groups of SBC implemented in the three models, for the two MILP solvers, resulting in 432 optimization runs, see Section 5.2. While in the second, the 24 instances and the three base models are solved by both MILP solvers using different levels of symmetry breaking cuts available in the MILP solvers, resulting in 504 additional optimization runs, see Section 5.3.

The overall results obtained from these optimization runs are summarized by using performance profiles (Dolan and Moré, 2002). These profiles provide a graphical methodology to compare the computational performance of the different models with and without the SBC. The profiles are based on the computational time required to solve the problem to optimality. Better performances correspond to a) a higher value of the cumulative distribution profile for $\tau = 1$; and b) the cumulative distribution profile curves closer to 1 for $\tau > 1$. Appendix C provides additional details about the profiles construction.

5.1. Performance of the base models

Tables 4, 5, and 6 present the performance and model statistics for the base models. These tables provide information about the size of the models, the optimal solution value of the linear relaxation models, the optimal solution value, and the corresponding optimality gap obtained with CPLEX and GUROBI.

Regarding the size of the models, Model III is the most compact model in terms of number of equations and nonzero elements. Model II is tighter than Model I, due to the constraints proposed by Ostrowski et al. (2012). Model III is the tightest model as it can be seen by comparing the optimal solution of the linear relaxation models. For example, for the same instances with 100 generators, the values of $cost^R$ obtained are \$5,691,587, \$5,694,777, and \$5,722,792, for Models I, II, and III, respectively.

In terms of computational performance, most of the instances cannot be solved to optimality within the time limit of 7200s. However, in all instances the solvers terminate with very low optimality gaps. Which as discussed in Section 3, it is explained by the fact that the linear relaxations of the three base models are very tight. In some additional tests, for example the instance with 35 generators from Case 2 cannot be solved to optimality even after 10 hours of computational time as shown in Table 1.

5.2. Analysis of the impact of SBC by model and solver

Figures 1, 2, and 3 present the relative computational performance obtained with CPLEX for the Models I, II, and II, without and with SBC. While Figures 4, 5, and 6 show the results obtained with GUROBI. In these figures, the performance profiles are based on the CPU times using: a) each of the three groups of SBC (S1, S2, S3); and b) without SBC (Model I, II, III); where each profile is built over the 24 instances. A qualitative analysis of these performance profiles indicates a clear-cut difference between using one of the SBC groups S1, S2, and S3 and the models without SBC. The results also show that the superior performance of the models with SBC is independent of the model, and MILP solver used.

A detailed analysis of Figures 1, 2, 3 indicates that with CPLEX, the following combinations:

1. Model I with SBC S1;
2. Model II with SBC S1;

Table 4: Case 1. Model statistics and results for the base models. CPU time limit set to 7,200s.

Model	NG	$cost^R$ (\$)	CPLEX *cost* (\$)	CPLEX G (%)	GUROBI *cost* (\$)	GUROBI G (%)	EQ	Var	0-1 Var
I	20	1,042,339	1,053,840	0.000	1,053,840	0.000	9,829	6,721	1,420
II	20	1,042,339	1,053,840	0.000	1,053,840	0.000	10,789	6,721	1,420
III	20	1,044,324	1,053,840	0.000	1,053,840	0.041	7,469	6,721	1,420
I	30	1,567,454	1,576,210	0.041	1,576,210	0.053	14,809	10,081	2,080
II	30	1,567,454	1,576,210	0.047	1,576,210	0.053	16,249	10,081	2,080
III	30	1,569,480	1,576,210	0.062	1,576,210	0.127	11,219	10,081	2,080
I	40	2,101,815	2,113,985	0.067	2,113,985	0.048	19,789	13,441	2,740
II	40	2,101,881	2,113,985	0.076	2,113,985	0.072	21,709	13,441	2,740
III	40	2,104,623	2,113,985	0.262	2,113,985	0.201	14,969	13,441	2,740
I	50	2,642,354	2,652,629	0.073	2,652,629	0.072	24,769	16,801	3,390
II	50	2,642,354	2,652,629	0.078	2,652,629	0.093	27,169	16,801	3,390
III	50	2,646,650	2,652,629	0.100	2,652,629	0.184	18,709	16,801	3,390
I	60	3,192,798	3,200,281	0.052	3,200,281	0.049	29,769	20,161	4,070
II	60	3,192,844	3,200,281	0.053	3,200,281	0.054	32,649	20,161	4,070
III	60	3,194,927	3,200,281	0.046	3,200,281	0.047	22,499	20,161	4,070
I	70	3,764,852	3,782,767	0.019	3,782,767	0.023	34,769	23,521	4,750
II	70	3,766,735	3,782,767	0.020	3,782,767	0.017	38,129	23,521	4,750
III	70	3,772,474	3,782,767	0.022	3,782,767	0.026	26,289	23,521	4,750
I	80	4,354,095	4,389,035	0.024	4,389,035	0.025	39,789	26,881	5,450
II	80	4,357,288	4,389,035	0.023	4,389,035	0.022	43,399	26,881	5,450
III	80	4,371,956	4,389,035	0.036	4,389,035	0.046	30,359	26,881	5,450
I	90	4,977,737	5,018,323	0.027	5,018,323	0.027	44,809	30,241	6,150
II	90	4,981,469	5,018,323	0.023	5,018,323	0.027	48,669	30,241	6,150
III	90	4,998,632	5,018,323	0.019	5,018,324	0.018	34,429	30,241	6,150
I	100	5,691,587	5,734,538	0.028	5,734,538	0.026	49,829	33,601	6,850
II	100	5,694,777	5,734,538	0.019	5,734,538	0.024	53,939	33,601	6,850
III	100	5,722,792	5,734,538	0.027	5,734,538	0.017	38,499	33,601	6,850

NG - Number of generators, $cost^R$ - Objective function value of the linear relaxation, G (%) - Relative optimality gap, EQ - Number of equations, Var - total number of variables, 0-1 Var - Number of binary variables.

Table 5: Case 2. Model statistics and results for the base models. CPU time limit set to 7,200s.

			CPLEX		GUROBI				
Model	NG	$cost^R$ ($)	*cost* ($)	G (%)	*cost* ($)	G (%)	EQ	Var	0-1 Var
I	28	3,807,562	3,831,110	0.000	3,831,110	0.000	13,794	9,409	1,970
II	28	3,808,180	3,831,110	0.000	3,831,110	0.000	15,138	9,409	1,970
III	28	3,813,757	3,831,110	0.000	3,831,110	0.000	10,472	9,409	1,970
I	35	4,785,060	4,814,267	0.008	4,814,267	0.004	17,231	11,761	2,450
II	35	4,785,105	4,814,267	0.004	4,814,267	0.005	18,888	11,761	2,450
III	35	4,794,574	4,814,267	0.039	4,814,267	0.053	13,090	11,761	2,450
I	44	5,094,546	5,122,977	0.025	5,122,977	0.022	21,725	14,785	3,057
II	44	5,094,814	5,122,977	0.014	5,122,977	0.013	23,768	14,785	3,057
III	44	5,101,711	5,122,977	0.002	5,122,977	0.039	16,538	14,785	3,057
I	45	4,767,768	4,791,616	0.000	4,791,616	0.008	22,259	15,121	3,130
II	45	4,767,905	4,791,616	0.000	4,791,616	0.000	24,350	15,121	3,130
III	45	4,774,074	4,791,616	0.000	4,791,616	0.006	16,956	15,121	3,130
I	49	5,363,159	5,390,419	0.013	5,390,419	0.017	24,213	16,465	3,383
II	49	5,364,229	5,390,419	0.010	5,390,419	0.017	26,542	16,465	3,383
III	49	5,369,748	5,390,419	0.007	5,390,419	0.009	18,359	16,465	3,383
I	50	4,379,393	4,399,862	0.001	4,399,862	0.002	24,811	16,801	3,435
II	50	4,379,410	4,399,862	0.000	4,399,862	0.004	27,096	16,801	3,435
III	50	4,387,630	4,399,862	0.000	4,399,862	0.009	18,916	16,801	3,435
I	51	5,144,746	5,174,282	0.009	5,174,282	0.014	25,240	17,137	3,535
II	51	5,145,443	5,174,282	0.013	5,174,282	0.015	27,527	17,137	3,535
III	51	5,153,441	5,174,282	0.004	5,174,282	0.010	19,304	17,137	3,535
I	51	5,805,112	5,834,913	0.007	5,834,913	0.013	25,184	17,137	3,564
II	51	5,805,169	5,834,913	0.015	5,834,913	0.004	27,540	17,137	3,564
III	51	5,814,092	5,834,913	0.020	5,834,913	0.032	19,205	17,137	3,564
I	52	5,593,260	5,622,215	0.010	5,622,215	0.015	25,708	17,473	3,596
II	52	5,594,423	5,622,215	0.013	5,622,215	0.014	28,089	17,473	3,596
III	52	5,600,318	5,622,215	0.016	5,622,215	0.019	19,596	17,473	3,596
I	54	5,044,723	5,072,047	0.019	5,072,047	0.020	26,758	18,145	3,729
II	54	5,045,305	5,072,047	0.015	5,072,047	0.018	29,212	18,145	3,729
III	54	5,055,460	5,072,047	0.015	5,072,047	0.027	20,425	18,145	3,729
I	65	6,178,619	6,211,472	0.019	6,211,548	0.023	32,189	21,841	4,440
II	65	6,179,546	6,211,472	0.019	6,211,472	0.021	35,286	21,841	4,440
III	65	6,191,252	6,211,472	0.023	6,211,472	0.022	24,368	21,841	4,440

NG - Number of generators, $cost^R$ - Objective function value of the linear relaxation, G (%) - Relative optimality gap, EQ - Number of equations, Var - total number of variables, 0-1 Var - Number of binary variables.

Table 6: Case 2. Model statistics and results for the base models. CPU time limit set to 7,200s. (Cont.)

			CPLEX		GUROBI				
Model	NG	$cost^R$ (\$)	*cost* (\$)	G (%)	*cost* (\$)	G (%)	EQ	Var	0-1 Var
I	75	7,182,070	7,221,361	0.008	7,221,396	0.008	37,126	25,201	5,161
II	75	7,183,632	7,221,361	0.008	7,221,361	0.012	40,611	25,201	5,161
III	75	7,197,588	7,221,361	0.014	7,221,361	0.015	28,232	25,201	5,161
I	85	8,634,821	8,676,655	0.013	8,676,655	0.013	42,029	28,561	5,903
II	85	8,635,246	8,676,655	0.020	8,676,655	0.018	45,902	28,561	5,903
III	85	8,645,865	8,676,655	0.019	8,676,655	0.014	32,083	28,561	5,903
I	95	10,475,089	10,525,890	0.011	10,525,890	0.010	46,885	31,921	6,578
II	95	10,475,630	10,525,892	0.012	10,525,890	0.012	51,376	31,921	6,578
III	95	10,486,630	10,525,890	0.010	10,525,892	0.014	35,580	31,921	6,578
I	100	11,622,782	11,681,603	0.024	11,681,603	0.024	49,310	33,601	6,966
II	100	11,623,117	11,681,603	0.025	11,681,603	0.026	53,926	33,601	6,966
III	100	11,639,182	11,681,603	0.032	11,681,603	0.030	37,568	33,601	6,966

NG - Number of generators, $cost^R$ - Objective function value of the linear relaxation, G (%) - Relative optimality gap, EQ - Number of equations, Var - total number of variables, 0-1 Var - Number of binary variables.

3. Model III with SBC S3;

have the highest probabilities of solving any given problem, namely 45.8%, 50.0%, and 41.7%, respectively, as it can be seen by the intersection of the profiles with the y-axis for $\tau = 1$.

A similar analysis for GUROBI shows that the combinations:

1. Model I with SBC S2 or S3;
2. Model II with SBC S2;
3. Model III with SBC S3;

have the highest probabilities of solving any given problem, namely 29.2%, 41.7%, and 29.2%, respectively, as it can be seen by the intersection of the profiles with the y-axis for $\tau = 1$, see Figures 4, 5, 6.

These results clearly show that there is not a group of SBC that outperforms the other SBC for all models and solvers. Furthermore, the effectiveness of the SBC depends on the Model and the MILP solver used. A thorough analysis of, for example, Figure 1, shows that even if the SBC S1 has the highest probability of 45.8% of being the best SBC to solve any problem, the SBC S2 still has the probability of 25.0% of being the best. The probability of 25.0% indicates that S2 would be a better choice for some cases.

Analyzing the structure of the groups of symmetry breaking cuts, the SBC S1 only considers the Equation 12, which is commonly used in the literature. While SBC S2 and S3 encompass a set of constraints derived in this work. The results presented show that in four combinations of model and solver out of the six combinations, the SBC S2 and S3 improve the performance of S1. This comparison demonstrates the value of the additional constraints included in S2 and S3.

Tables 7 to 8 present detailed computational results for the nine instances from Case 1 and the 15 instances of Case 2. The results are shown for the base models (columns with the header BASE) and the corresponding modified models including the SBC group S3 (columns with the header SBC). For both cases, a general overview of these results indicates that the models including the SBC require the evaluation of a significant lower number of nodes, and can close the optimality gap in a significant lower CPU time. For example, focusing first in Case 1 with the S3, and analyzing the results obtained with Model III for the instances with 60 and 70 generators. With the base model after two hours, the MILP solver evaluated 2,327,245 and 500,239 nodes, respectively, and it does not close the optimality gap. While with the SBC, for the same instances, the MILP solver only requires 581,685 and 154,102 nodes to close the gap in 5,849s and 1,605s, (see Table 7).

For Case 2 with S3, the impact of using the SBC is demonstrated by comparing the results presented in Table 1 with those in Table 8 for the instance with 35 generators. For this instance, the MILP solver cannot close the optimality gap within 10 hours with the three models. While when the SBC are considered in these models the solver terminates in 58s, 74s, and 17s, respectively for Models I, II, and III, see Table 8. The superior performance of the three Models with SBC for the instances with less or equal than 54 generators is evident since all instances are solved to optimality with CPU times significantly lower than 7,200s. Note that in Case 2, the Models I and III with SBC can close the optimality gap within the

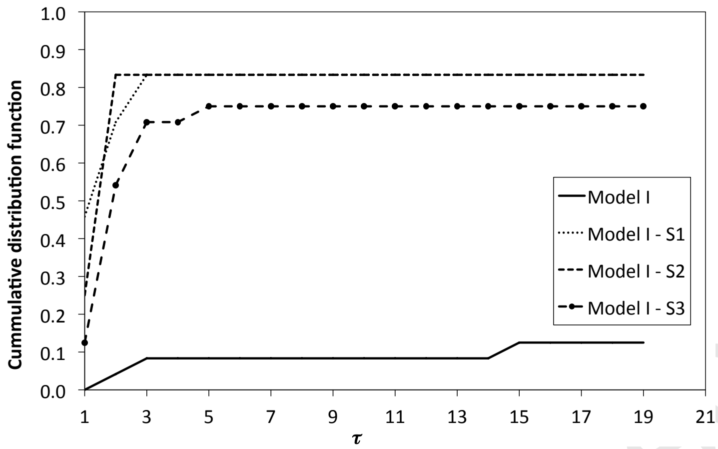

Figure 1: Comparison between Model I without SBC and Model I with different SBC. Profiles obtained for CPLEX, over the 24 instances.

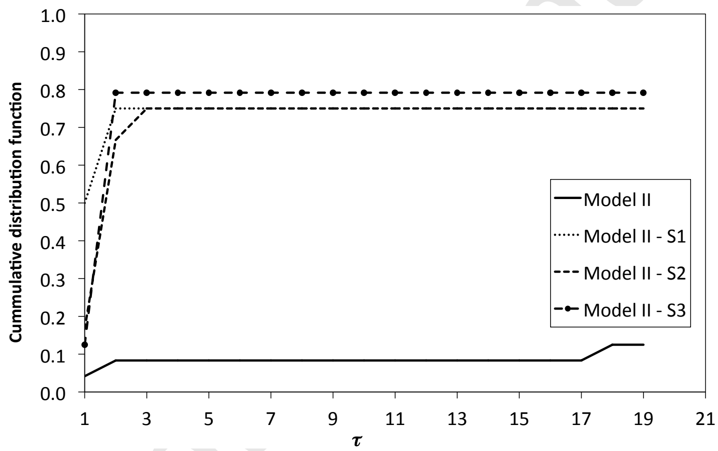

Figure 2: Comparison between Model II without SBC and Model II with different SBC. Profiles obtained for CPLEX, over the 24 instances.

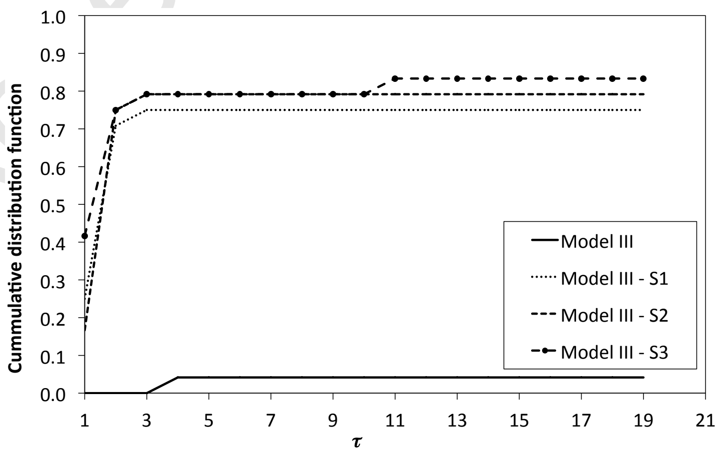

Figure 3: Comparison between Model III without SBC and Model III with different SBC. Profiles obtained for CPLEX, over the 24 instances.

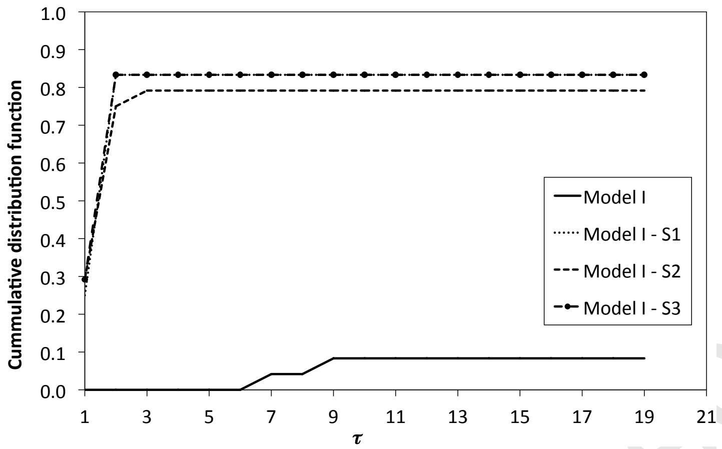

Figure 4: Comparison between Model I without SBC and Model I with different SBC. Profiles obtained for GUROBI, over the 24 instances.

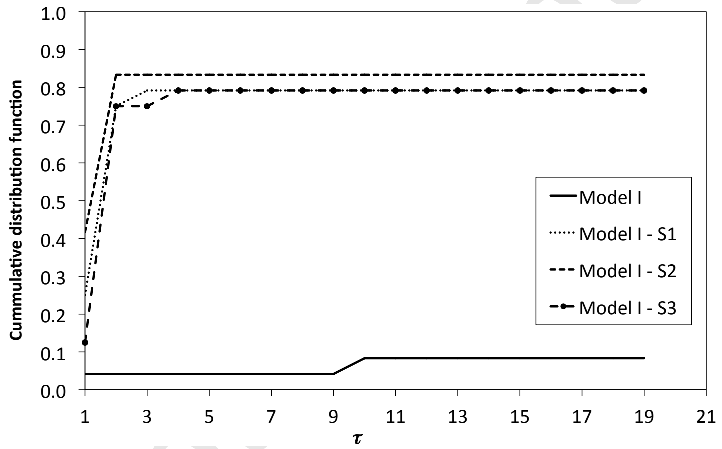

Figure 5: Comparison between Model II without SBC and Model II with different SBC. Profiles obtained for GUROBI, over the 24 instances.

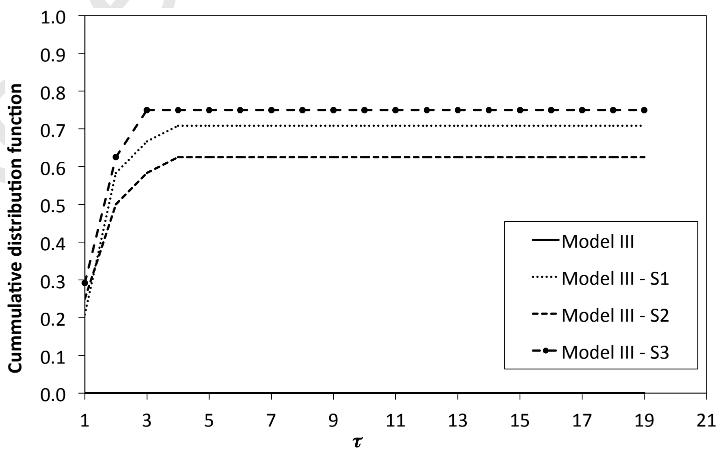

Figure 6: Comparison between Model III without SBC and Model III with different SBC. Profiles obtained for GUROBI, over the 24 instances.

time limit for the two larger problems with 95 and 100 generators.

Table 7: Case 1. Comparison between the base models without (BASE) and with SBC S3. Bold means solved to optimality gap of 0.0%.

		BASE	S3	BASE	S3	BASE	S3	BASE	S3
Model	NG	*cost* ($)	*cost* ($)	# Nodes	# Nodes	G (%)	G (%)	T (s)	T (s)
I	20	1,053,840	1,053,840	2,527	593	0.000	0.000	**18**	**20**
I	30	1,576,210	1,576,210	553,396	751	0.041	0.000	7,209	**33**
I	40	2,113,985	2,113,985	323,976	18,785	0.067	0.000	7,215	**477**
I	50	2,652,629	2,652,629	294,097	21,359	0.073	0.000	7,274	**865**
I	60	3,200,281	3,200,281	1,477,583	289,734	0.052	0.010	7,201	7,202
I	70	3,782,767	3,782,767	308,453	62,995	0.019	0.000	OM	**2,956**
I	80	4,389,035	4,389,035	1,103,586	187,279	0.024	0.012	7,201	7,243
I	90	5,018,323	5,018,354	1,099,971	169,425	0.027	0.011	7,205	7,240
I	100	5,734,538	5,734,538	1,668,987	901,047	0.028	0.025	7,206	7,201
II	20	1,053,840	1,053,840	2,451	674	0.000	0.000	**13**	**21**
II	30	1,576,210	1,576,210	480,490	692	0.047	0.000	7,224	**39**
II	40	2,113,985	2,113,985	348,118	16,134	0.076	0.000	7,230	**511**
II	50	2,652,629	2,652,629	265,350	15,956	0.078	0.000	7,285	**640**
II	60	3,200,281	3,200,281	565,891	236,553	0.053	0.000	7,237	**7,088**
II	70	3,782,767	3,782,767	347,334	36,240	0.020	0.000	OM	**1,493**
II	80	4,389,035	4,389,035	1,080,710	190,163	0.023	0.010	7,203	7,274
II	90	5,018,323	5,018,323	866,061	132,892	0.023	0.014	7,204	7,251
II	100	5,734,538	5,734,569	1,694,206	392,555	0.019	0.030	7,205	7,214
III	20	1,053,840	1,053,840	55,360	2,583	0.000	0.000	**58**	**23**
III	30	1,576,210	1,576,210	797,021	2,630	0.062	0.000	7,205	**42**
III	40	2,113,985	2,113,985	331,341	36,916	0.262	0.000	7,259	**705**
III	50	2,652,629	2,652,629	347,345	27,990	0.100	0.000	7,237	**567**
III	60	3,200,281	3,200,281	2,327,245	581,658	0.046	0.000	7,200	**5,849**
III	70	3,782,767	3,782,767	500,239	154,102	0.022	0.000	7,221	**1,605**
III	80	4,389,035	4,389,038	2,172,764	337,592	0.036	0.026	7,201	7,244
III	90	5,018,323	5,018,323	2,006,172	261,108	0.019	0.031	7,206	7,258
III	100	5,734,538	5,734,538	2,793,411	1,810,709	0.027	0.032	7,205	7,201

NG - Number of generators, G - Relative optimality gap, T - CPU time, OM - Out of memory.

The results presented in Tables 4 to 8 show that the same value of the objective function is obtained with the models without and with SBC. There is no improvement of the objective function using the models with SBC. This is explained by the fact that the MILP solvers use good heuristics to find integer solutions. However, despite the fact that the MILP solvers can find a good integer solution and the linear relaxations of the models are very tight, the MILP solvers cannot close the optimality gap. As described in Section 3, this was the motivation to develop the SBC.

Table 8: Case 2. Comparison between the base models without (BASE) and with SBC S3. Bold means solved to optimality gap of 0.0%.

		BASE	S3	BASE	S3	BASE	S3	BASE	S3
Model	NG	*cost* ($)	*cost* ($)	# Nodes	# Nodes	G (%)	G (%)	T (s)	T (s)
I	28	3,831,110	3,831,110	2,732	362	0.000	0.000	**50**	**28**
I	35	4,814,267	4,814,267	3,497,483	2,334	0.008	0.000	7,201	**58**
I	44	5,122,977	5,122,977	446,394	29,590	0.025	0.000	7,235	**478**
I	45	4,791,616	4,791,616	441,706	5,853	0.000	0.000	**1,497**	**116**
I	49	5,390,419	5,390,419	3,055,274	15,389	0.013	0.000	7,201	**198**
I	50	4,399,862	4,399,862	3,738,738	8,689	0.001	0.000	7,200	**197**
I	51	5,174,282	5,174,282	2,892,161	30,739	0.009	0.000	7,201	**307**
I	51	5,834,913	5,834,913	2,004,566	33,104	0.007	0.000	7,201	**509**
I	52	5,622,215	5,622,215	624,621	53,289	0.010	0.000	7,238	**559**
I	54	5,072,047	5,072,047	511,397	114,652	0.019	0.000	7,263	**1,217**
I	65	6,211,472	6,211,521	954,716	178,693	0.019	0.026	7,201	7,205
I	75	7,221,361	7,221,361	1,572,372	87,919	0.008	0.000	7,204	**2,091**
I	85	8,676,655	8,676,655	1,500,497	626,222	0.013	0.002	7,201	7,200
I	95	10,525,890	10,525,890	1,332,077	124,059	0.011	0.000	7,209	**2,845**
I	100	11,681,603	11,681,603	207,311	165,830	0.024	0.000	OM	**7,037**
II	28	3,831,110	3,831,110	2,376	925	0.000	0.000	**53**	**38**
II	35	4,814,267	4,814,267	3,307,586	2,304	0.004	0.000	7,201	**74**
II	44	5,122,977	5,122,977	455,599	29,296	0.014	0.000	7,213	**399**
II	45	4,791,616	4,791,616	348,009	5,010	0.000	0.000	**1,008**	**89**
II	49	5,390,419	5,390,419	2,766,067	16,975	0.010	0.000	7,200	**306**
II	50	4,399,862	4,399,862	4,016,839	7,048	0.000	0.000	**6,482**	**213**
II	51	5,174,282	5,174,282	2,580,322	32,394	0.013	0.000	7,201	**354**
II	51	5,834,913	5,834,913	467,307	15,148	0.015	0.000	7,241	**249**
II	52	5,622,215	5,622,215	2,346,958	49,964	0.013	0.000	7,201	**655**
II	54	5,072,047	5,072,047	1,621,988	199,442	0.015	0.000	7,201	**1,858**
II	65	6,211,472	6,211,472	886,650	148,204	0.019	0.039	7,201	7,214
II	75	7,221,361	7,221,361	1,781,636	227,190	0.008	0.000	7,203	**5,527**
II	85	8,676,655	8,676,655	1,340,551	308,883	0.020	0.000	7,201	**3,678**
II	95	10,525,892	10,525,890	1,067,027	288,958	0.012	0.001	7,201	7,201
II	100	11,681,603	11,681,603	597,589	97,455	0.025	0.000	7,201	**5,639**
III	28	3,831,110	3,831,110	1,194,561	669	0.000	0.000	**823**	**8**
III	35	4,814,267	4,814,267	683,910	2,363	0.039	0.000	7,227	**17**
III	44	5,122,977	5,122,977	1,348,440	40,077	0.002	0.000	7,201	**260**
III	45	4,791,616	4,791,616	557,540	9,724	0.000	0.000	**707**	**43**
III	49	5,390,419	5,390,419	877,898	44,813	0.007	0.000	7,227	**315**
III	50	4,399,862	4,399,862	1,547,974	12,292	0.000	0.000	**1,712**	**76**
III	51	5,174,282	5,174,282	5,666,366	30,742	0.004	0.000	7,201	**102**
III	51	5,834,913	5,834,913	3,496,820	93,083	0.020	0.000	7,200	**811**
III	52	5,622,215	5,622,215	570,438	90,119	0.016	0.000	7,226	**481**
III	54	5,072,047	5,072,047	2,773,487	422,377	0.015	0.000	7,200	**1,255**
III	65	6,211,472	6,211,472	1,067,384	444,278	0.023	0.006	7,211	7,208
III	75	7,221,361	7,221,361	2,374,255	317,483	0.014	0.000	7,203	**2,209**
III	85	8,676,655	8,676,655	2,231,718	972,990	0.019	0.000	7,201	**3,646**
III	95	10,525,890	10,525,890	1,217,336	415,959	0.010	0.000	7,201	**4,434**
III	100	11,681,603	11,681,603	1,679,081	154,635	0.032	0.000	7,201	**4,195**

NG - Number of generators, G - Relative optimality gap, T - CPU time, OM - Out of memory.

5.3. Comparison between the SBC and the symmetry breaking methods available in MILP solvers

In this section, we present a comparison between the proposed SBC and the symmetry breaking methods available in CPLEX and GUROBI. For CPLEX, we have tested the default value, $\{-1\}$, and three levels of symmetry breaking cuts, $\{1, 3, 5\}$, where 5 is the most aggressive mode. While for GUROBI, in addition to the default value, $\{-1\}$, two levels of symmetry cuts were considered, $\{1, 2\}$, where 2 is the most aggressive mode. The test set is built over the 24 instances and the three models.

Figures 7, 8, and 9 show the performance profiles obtained with CPLEX for each model with the four modes of symmetry breaking cuts from CPLEX, and one group of the proposed SBC in this work. While Figures 10, 11, and 12 show the profiles obtained with GUROBI for each model with the three modes of symmetry breaking cuts from GUROBI, and one group of the proposed SBC.

These results show that the proposed SBC have a superior performance than any of the different levels of symmetry breaking cuts embedded in CPLEX or GUROBI. Note that for all models with CPLEX or GUROBI, the proposed SBC have a probability clearly above of 70% of being the best approach solving any given problem to optimality. While, on the other hand, the symmetry breaking cuts within the solvers are not at all efficient in closing the optimality gap. This performance is demonstrated by the small values of the cumulative distribution functions obtained by those methods within the solvers.

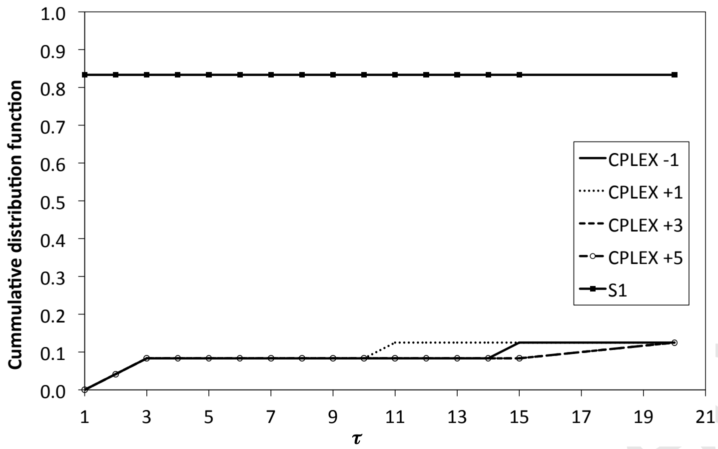

Figure 7: Comparison between Model I solved with the CPLEX symmetry breaking options $\{-1, 1, 3, 5\}$, and Model I with the SBC S1. Profiles obtained for CPLEX, over the 24 instances.

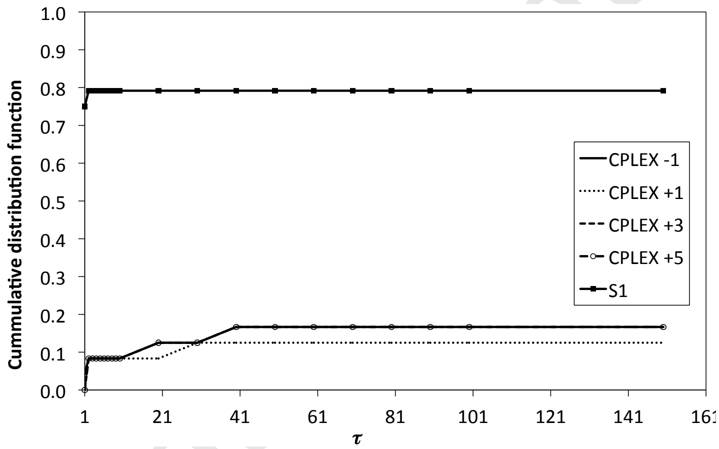

Figure 8: Comparison between Model II solved with the CPLEX symmetry breaking options $\{-1, 1, 3, 5\}$, and Model II with the SBC S1. Profiles obtained for CPLEX, over the 24 instances.

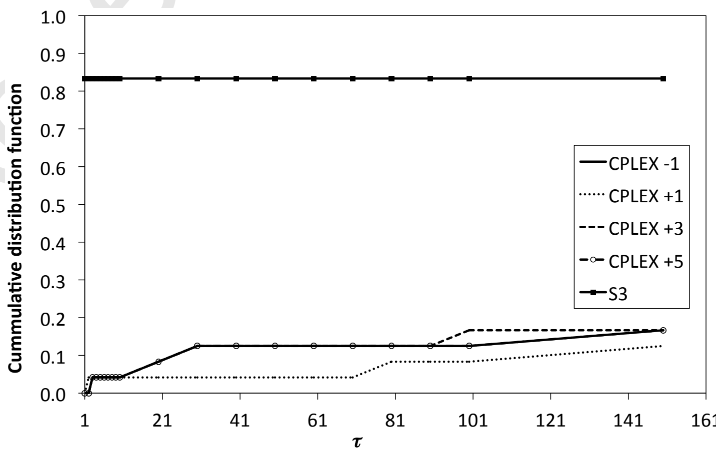

Figure 9: Comparison between Model III solved with the CPLEX symmetry breaking options $\{-1, 1, 3, 5\}$, and Model III with the SBC S3. Profiles obtained for CPLEX, over the 24 instances.

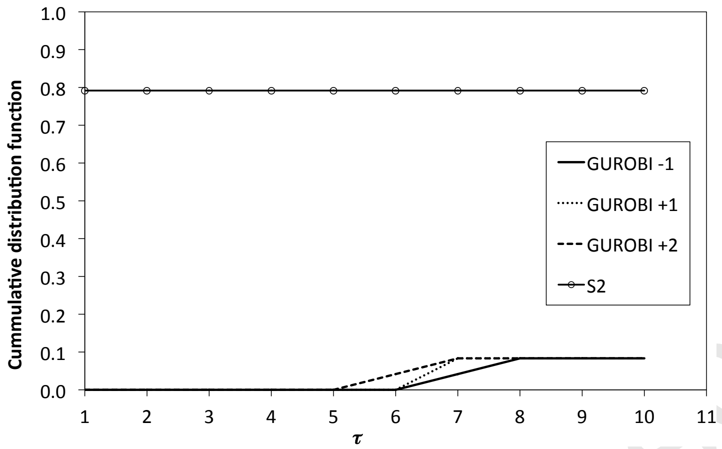

Figure 10: Comparison between Model I solved with the GUROBI symmetry breaking options $\{-1, 1, 2\}$, and Model I with the SBC S1. Profiles obtained for GUROBI, over the 24 instances.

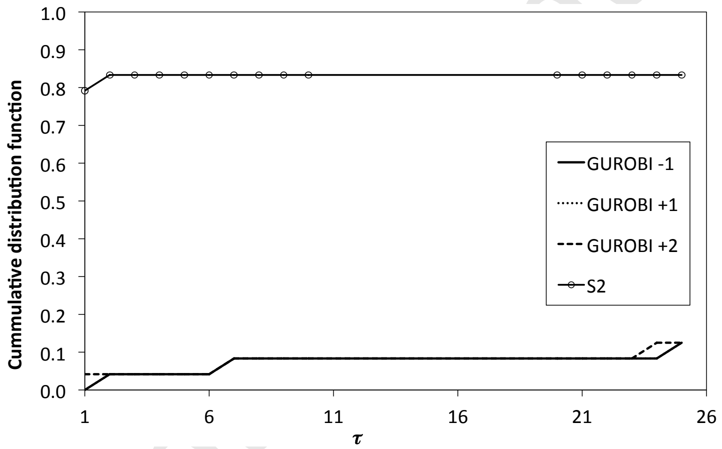

Figure 11: Comparison between Model II solved with the GUROBI symmetry breaking options $\{-1, 1, 2\}$, and Model II with the SBC S2. Profiles obtained for GUROBI, over the 24 instances.

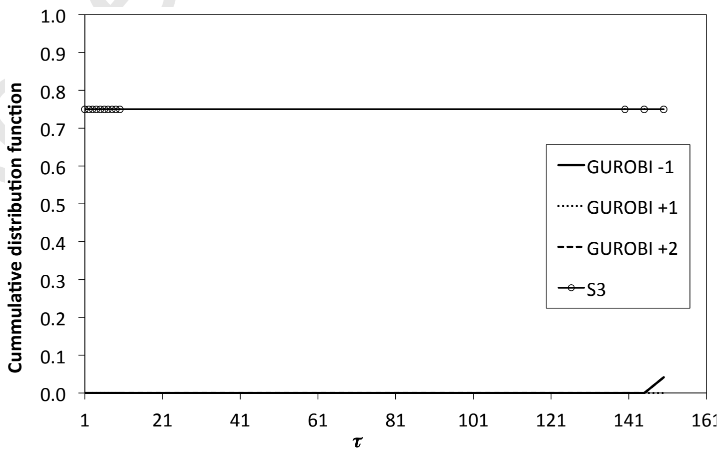

Figure 12: Comparison between Model III solved with the GUROBI symmetry breaking options $\{-1, 1, 2\}$, and Model III with the SBC S3. Profiles obtained for GUROBI, over the 24 instances.

6. Conclusions

The two main goals of this work are to demonstrate the impact of symmetry in the standard benchmark problems used to test UC MILP problems and to show the value of incorporating the SBC. Regarding the first goal, some motivating examples are used to identify specific cases of symmetry in UC solutions and to demonstrate the impact of the symmetry in the computational time required to prove the optimality of the solutions. Based on these findings, we propose three groups of SBC, and we discuss their limitations in terms of breaking the symmetry. The first group of SBC is commonly used in the literature while the other two groups of SBC encompass cuts derived in this work. The computational results demonstrate that for some instances, models, and solvers, the two later SBC improve the computational performance of the first group of SBC.

The results clearly indicate that the introduction of specific SBC in the studied models allows the MILP solvers to handle effectively a class of problems that cannot otherwise be solved at all or require extensive computational burden. The proposed SBC revealed to be more efficient than the symmetry breaking methods implemented in CPLEX and GUROBI. In fact, no significant improvement in the computational performance was achieved with different levels of aggressiveness of the symmetry breaking methods available from the solvers. The results obtained show that there is no improvement in the objective function by using the models with SBC. However, the inclusion of the SBC do not lead to a marginal improvement of the optimality gap for a given CPU time, but rather to a disruptive improvement consisting in closing the optimality gap within reasonable CPU times.

For larger problems the impact of the proposed SBC in the performance of the solvers decreases. This deterioration may be due to the existence of symmetry that is not broken with the proposed SBC, or due to the size of the problems that would require computers with faster CPUs.

However, in some real applications, not breaking symmetry might offer some advantages. First, there are applications where having available alternative solutions with an equal objective value is necessary for the decision maker. Note that currently, off-the-shelf MILP solvers provide the ability to generate solution pools during the B&B algorithm. These alternative solutions give the option to the decision maker to choose one solution from a pool of solutions with the same objective value. This pool of solutions is particularly interesting when the model does not capture all the features of the process. Another disadvantage is that the solver always commits first the generators with the lower index value within the set of indistinguishable generators, which might be considered unreasonable. However, this can be avoided by randomly assigning an order to the generators within the set of indistinguishable generators before the optimization. This assignment is equivalent to having the MILP solver picking one optimal solution from the possible options with an equal objective value.

The proposed SBC can be easily incorporated into UC MILP models for real generation systems with symmetry issues, without a significant impact on the size of the models. Symmetry breaking is a complement, and not an alternative, to compact and tight formulations. With the current state-of-the-art MILP solvers, such formulations may not necessarily mean that the objective function value will reflect monetary savings in dispatch costs or reserves. Within the same rationale, SBC may not lead by themselves to monetary savings. However, the combination of the three features of UC MILP models: compactness, tightness, and symmetry free, are instrumental to help efficiently the finding of optimal solutions in a short time and prove their optimality. Thus, reducing the risk of a sub-optimal solution due to computational time limitations, and providing added-value to any operation or negotiation requiring optimization results. On the other hand, from the perspective of benchmarking UC MILP models, the inclusion of SBC may, in fact, isolate and reveal the worth of compact and tight representations.

Acknowledgments

The first author acknowledges the support of the Center for Uncertainty Quantification in Computational Science & Engineering in KAUST. The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement n. PCOFUND-GA-2009-246542 and from the Foundation for

Science and Technology of Portugal.

Appendix A

In this appendix are described the three base models used in this work. A detailed description of the models can be found in the respective references indicated in the next subsections. However, from the models presented below, it is clear that the main characteristics of typical UC problems are used in this work.

6.1. Objective function linearization

The production costs are represented by a quadratic term of the variable $p_{i,t}$, which are approximated using a piecewise linear formulation, as used by Carrion and Arroyo (2006). This is a standard approximation that is also described in Williams (1999).

6.1.1. Model I and II

The production costs, $cp_{i,t}$ are given by

$$cp_{i,t} = a_i u_{i,t} + b_i p_{i,t} + c_i p_{i,t}^2, \quad \forall i, t, \tag{19}$$

where a_i, b_i, and c_i are the cost coefficients given in Table 9. The quadratic term $p_{i,t}^2$ is substituted by a piecewise linear approximation, whereby we replace $p_{i,t}^2$ by $\pi_{i,t}$ in the definition of $cp_{i,t}$,

$$cp_{i,t} = a_i u_{i,t} + b_i p_{i,t} + c_i \pi_{i,t}, \quad \forall i, t, \tag{20}$$

and the following piecewise linear formulation is used:

$$p_{i,t} = P_i^L u_{i,t} + \sum_k \gamma_{i,t,k}, \quad \forall i, t, \tag{21}$$

$$\pi_{i,t} = \left(P_i^L\right)^2 u_{i,t} + \sum_k \left(\frac{Y_{i,t,k}}{X_{i,t,k}} \gamma_{i,t,k}\right), \quad \forall i, t, \tag{22}$$

$$\gamma_{i,t,k} \leq X_{i,t,k}, \quad \forall i, t, k, \tag{23}$$

where $X_{i,t,k}$ and $Y_{i,t,k}$ denote the fixed length of the interval k of the partition scheme of the variable $p_{i,t}$, and the length of the interval k of the partition scheme of the variable $\pi_{i,t}$, respectively, and $\gamma_{i,t,k}$ is the power output in the partition k of unit i in period t. We used grids with three partitions with the same size between P_i^L and P_i^U.

6.1.2. Model III

The production costs in Model III are defined as

$$cp_{i,t} = a_i u_{i,t} + b_i \left(P_i^L u_{i,t} + pu_{i,t}\right) + c_i \pi_{i,t}, \quad \forall i, t, \tag{24}$$

and the piecewise linear formulation is given by the following equation

$$pu_{i,t} = \sum_k \gamma_{i,t,k}, \quad \forall i, t, \tag{25}$$

plus Equations (22) and (23).

6.2. Feasible region of operation of the generators

6.2.1. Model I

Model I is based on Ostrowski et al. (2012), which results from the contributions of different authors.

$$P_i^L u_{i,t} \leq p_{i,t} \leq \overline{p}_{i,t} \leq P_i^U u_{i,t}, \quad \forall i, t, \tag{26}$$

$$\overline{p}_{i,t} - P0_i \leq RU_i u_{i,t-1} + SU_i u_{i,t}^{up}, \quad \forall i, t = 1, \tag{27}$$

$$\overline{p}_{i,t} - p_{i,t-1} \leq RU_i u_{i,t-1} + SU_i u_{i,t}^{up}, \quad \forall i, t \geq 2, \tag{28}$$

$$P0_i - p_{i,t} \leq RD_i u_{i,t} + SD_i u_{i,t}^{dn}, \quad \forall i, t = 1, \tag{29}$$

$$p_{i,t-1} - p_{i,t} \leq RD_i u_{i,t} + SD_i u_{i,t}^{dn}, \quad \forall i, t \geq 2, \tag{30}$$

$$\overline{p}_{i,t} \leq P_i^U u_{i,t+1} + SD_i (u_{i,t} - u_{i,t+1}) \quad \forall i, t < |T|. \tag{31}$$

$$\sum_{tt \geq t - UT_i + 1, tt \leq t} u_{i,tt}^{up} \leq u_{i,t} \quad \forall i, t \geq LM_i + 1, \tag{32}$$

$$u_{i,t} + \sum_{tt \geq t - DT_i + 1, tt \leq t} u_{i,tt}^{dn} \leq 1 \quad \forall i, t \geq FM_i + 1, \tag{33}$$

$$cu_{i,t} \geq u_{i,t}^{up} HS_i, \quad \forall i, t. \tag{34}$$

$$cu_{i,t} \geq \left(u_{i,t} - \sum_{tt \geq t-(DT_i+T_i^c+1)}^{t-1} u_{i,tt} \right) CS_i, \quad \forall i, t > DT_i + T_i^c + 1. \tag{35}$$

$$cu_{i,t} \geq \left(u_{i,t} - \sum_{tt<t} u_{i,tt} \right) CS_i, \quad \forall i, T_i^I < 0, \left(DT_i + T_i^c + T_i^I + 1\right) < t \leq \left(DT_i + T_i^c + 1\right). \tag{36}$$

$$cd_{i,t} \geq u_{i,t}^{dn} DC_i \quad \forall i, t. \tag{37}$$

$$1 - u_{i,t} + u_{i,t}^{up} - u_{i,t}^{dn} = 0, \quad \forall i, t = 1, T^I > 0, \tag{38}$$

$$- u_{i,t} + u_{i,t}^{up} - u_{i,t}^{dn} = 0, \quad \forall i, t = 1, T^I < 0, \tag{39}$$

$$u_{i,t-1} - u_{i,t} + u_{i,t}^{up} - u_{i,t}^{dn} = 0, \quad \forall i, t > 1. \tag{40}$$

The above model for the thermal unit involves: a) bounds on power output, Equations (26) to (31), see Ostrowski et al. (2012); b) minimum up and down time limits, Equations (32), and (33), see Rajan and Takriti (2005); c) modeling of starting, and shutdown costs, Equations (35) to (36), see Arroyo and Conejo (2000); Morales-Espana et al. (2013); Nowak and Romisch (2000); and d) logical relations between the different states of the unit, Eqs. (38) to (40).

6.2.2. Model II

Model III involves all the equations from Model II, plus some selected ramp rate constraints from Ostrowski et al. (2012).

$$p_{i,t} \leq P^U u_{i,t+K_t} + \sum_{k \in K_t} \left[(SD_i + (k-1)RD_i)\, u_{i,t+k}^{dn} \right] - \sum_{k \in K_t} P^U u_{i,t+k}^{up}, \quad \forall i, t, \tag{41}$$

$$P0_i - p_{i,t} \leq RD_i u_{i,t} + SD_i u_{i,t}^{dn} - \left(RD_i - SU_i + P_i^L\right) U0_i^{up} - \left(RD_i + P_i^L\right) u_{i,t}^{up}, \quad \forall i, t = 1, \tag{42}$$

$$p_{i,t-1} - p_{i,t} \leq RD_i u_{i,t} + SD_i u_{i,t}^{dn} - \left(RD_i - SU_i + P_i^L\right) u_{i,t-1}^{up} - \left(RD_i + P_i^L\right) u_{i,t}^{up}, \quad \forall i, t \geq 2. \tag{43}$$

$$p_{i,t-2} - p_{i,t} \leq 2RD_i u_{i,t} + SD_i u^{dn}_{i,t-1} + (SD_i + RD_i)\, u^{dn}_{i,t} - \left(2RD_i - SU_i + P^L_i\right) u^{up}_{i,t-2}$$
$$- \left(2RD_i + P^L_i\right) u^{up}_{i,t-1} - \left(2RD_i + P^L_i\right) u^{up}_{i,t}, \quad \forall i, t \geq 3, \tag{44}$$

$$P0_i - p_{i,t} \leq 2RD_i u_{i,t} + SD_i u^{dn}_{i,t-1} + (SD_i + RD_i)\, u^{dn}_{i,t} - \left(2RD_i - SU_i + P^L_i\right) u0^{up}_i$$
$$- \left(2RD_i + P^L_i\right) u^{up}_{i,t-1} - \left(2RD_i + P^L_i\right) u^{up}_{i,t}, \quad \forall i, t = 2. \tag{45}$$

$$\overline{p}_{i,t} - P0_i \leq RU_i u_{i,t} - P^L_i u^{dn}_{i,t} - \left(RU_i - SD_i + P^L_i\right) u^{dn}_{i,t+1} + (SU_i - RU_i)\, u^{up}_{i,t}, \quad \forall i, t = 1, \tag{46}$$

$$\overline{p}_{i,t} - p_{i,t-1} \leq RU_i u_{i,t} - P^L_i u^{dn}_{i,t} - \left(RU_i - SD_i + P^L_i\right) u^{dn}_{i,t+1} + (SU_i - RU_i)\, u^{up}_{i,t}, \quad \forall i, t \geq 2. \tag{47}$$

6.2.3. Model III

Model III is a recent model proposed in Morales-Espana et al. (2013). This model features a tight and compact formulation, which is based on: a) an alternative formulation for the approximation of the startup costs with a tight definition; b) the power output is modeled by the variable $pu_{i,t}$.

$$\sum_i \left(P^L_i u_{i,t} + pu_{i,t}\right) = D_t, \quad \forall t, \tag{48}$$

$$\sum_i r_{i,t} \geq SR_t, \quad \forall t. \tag{49}$$

$$pu_{i,t} + r_{i,t} \leq \left(P^U_i - P^L_i\right) u_{i,t} - \left(P^U_i - SU_i\right) u^{up}_{i,t}, \quad \forall i \in I^1, t, \tag{50}$$

$$pu_{i,t} + r_{i,t} \leq \left(P^U_i - P^L_i\right) u_{i,t} - \left(P^U_i - SD_i\right) u^{dn}_{i,t+1}, \quad \forall i \in I^1, t, \tag{51}$$

$$pu_{i,t} + r_{i,t} \leq \left(P^U_i - P^L_i\right) u_{i,t} - \left(P^U_i - SU_i\right) u^{up}_{i,t} - \left(P^U_i - SD_i\right) u^{dn}_{i,t+1}, \quad \forall i \notin I^1, t, \tag{52}$$

$$pu_{i,t} + r_{i,t} - pu_{i,t-1} \leq RU_i, \quad \forall i, t, \tag{53}$$

$$- pu_{i,t} + pu_{i,t-1} \leq RD_i, \quad \forall i, t. \tag{54}$$

$$\delta_{ilt} = \sum_{t' \in N_{ilt}} u^{dn}_{i,t'}, \quad \forall i, l < |L|, t \geq T^l_{il+1}, \tag{55}$$

$$\sum_{l \in L} \delta_{ilt} = u^{up}_{i,t}, \quad \forall i, t, \tag{56}$$

This model for the thermal unit involves: a) an alternative formulation for the energy balance and minimum reserves requirement; a) bounds on power output, Equations (50) to (54); b) minimum up and down time limits, Equations (32), and (33), see Rajan and Takriti (2005); c) the derivation of the startup costs formulation requires a detailed explanation, which is out of the scope of this work. The interested reader is directed to the original work that contains a thoroughly explanation, see Morales-Espana et al. (2013). The startup costs are captured by the two Equations (55) and (56).

Appendix B

In this appendix are presented the tables with the data used in both case studies. This data is based in Ostrowski et al. (2012) and Kazarlis et al. (1996).

Table 9: Base data for unique generators Kazarlis et al. (1996).

Gen	P_i^L (MW)	P_i^U (MW)	UT_i (h)	DT_i (h)	T_i^c (h)	T_i^I (h)	a_i (\$/h)	b_i (\$/MWh)	c_i (\$/(MW^{2}h))	HS_i (\$/h)	CS_i (\$/h)
1	150	455	8	8	5	8	1000	16.19	0.00048	4500	9000
2	150	455	8	8	5	8	970	17.26	0.00031	5000	10000
3	20	130	5	5	4	-5	700	16.6	0.002	550	1100
4	20	130	5	5	4	-5	680	16.5	0.00211	560	1120
5	25	162	6	6	4	-6	450	19.7	0.00398	900	1800
6	20	80	3	3	2	-3	370	22.26	0.00712	170	340
7	25	85	3	3	2	-3	480	27.74	0.00079	260	520
8	10	55	1	1	0	-1	660	25.92	0.00413	30	60
9	10	55	1	1	0	-1	665	27.27	0.00222	30	60
10	10	55	1	1	0	-1	670	27.79	0.00173	30	60

Table 10: Power demand. Case 1 units in MW, and Case 2 in (%).

Time	1	2	3	4	5	6	7	8	9	10	11	12
Case 1	700	750	850	950	1000	1100	1150	1200	1300	1400	1450	1500
Case 2	0.71	0.65	0.62	0.6	0.58	0.58	0.6	0.64	0.73	0.8	0.82	0.83
Time	13	14	15	16	17	18	19	20	21	22	23	24
Case 1	1400	1300	1200	1050	1000	1100	1200	1400	1300	1100	900	800
Case 2	0.82	0.8	0.79	0.79	0.83	0.91	0.9	0.88	0.85	0.84	0.79	0.74

Appendix C

The performance profiles presented are based on the methodology presented in Dolan and Moré (2002), who have proposed an alternative graphical comparison methodology to support the benchmarking of optimization software. This type of profiles are nowadays a standard alternative to other comparison techniques based on averages and standard deviations, or other statistical indicators.

The performance profiles represent for a given set of results, P, the cumulative distribution function, $p_s(\tau)$, of the performance ratios between the computational time of the instance p solved with the solver s and the minimum of the computational times taken by the other solvers for the same instance. Thus the performance ratio is given by

$$\rho_{p,s} = \frac{t_{p,s}}{\min\{t_{p,s} : 1 \leq s \leq n_s\}}, \quad \forall p, s \tag{57}$$

and the cumulative performance profile is defined as

$$p_s(\tau) = \frac{1}{n_p} \text{size}\left\{p \in P : \rho_{p,s} \leq \tau\right\}, \quad \forall s \tag{58}$$

where $t_{p,s}$ is the computational time that solver s needed to solve problem p, n_s is the number of solvers, n_p is the number of instances, and τ denotes the ratio of the time factor of interest. The performance ratios with $\tau = 1$ correspond to the solvers with the minimum computational time, and therefore, $p_s(1)$ is the probability of the solver s to have the best performance on any given problem (Dolan and Moré, 2002).

Nomenclature

Sets

I	Set of generating units
I^1	Set of generating units with $UT_i = 1$
K	Set of partitions used to approximate the production costs
L	Set of partitions used to calculate startup costs
M_i	Set of indistinguishable generators of i
T	Set of time periods

Parameters

a_i, b_i, c_i	Production cost function coefficients for unit i (\$/h)
CS_i	Cold start-up cost of unit i (\$/h)
D_t	Power load demand for period t (MW)
DM_i	Number of periods unit i must be off at the beginning of the time horizon
DC_i	Shut-down cost (\$)
DT_i	Minimum down time of unit i (h)
FM_i	Minimum number of periods a unit i must be off at the beginning of the time horizon
HS_i	Hot start cost of unit i (\$/h)
LM_i	Minimum number of periods a unit i must be on at the beginning of the time horizon
P_i^L	Minimum power output of unit i (MW)
P_i^U	Maximum power output of unit i (MW)
$P0_i$	Power produced at t=0 by unit i (MW)
RD_i	Maximum ramp-down rate of unit i (MW)
RU_i	Maximum ramp-up rate of unit i (MW)
SD_i	Maximum shutdown rate of unit i (MW)
SR_t	Spinning reserve for period t (MW)
SU_i	Maximum start-up rate of unit i (MW)
U_i	Number of periods unit i must be on at the beginning of the time horizon
$U0_i$	Initial state of unit i {on,off}={1,0}
UT_i	Minimum up time of unit i (h)
T_i^c	Cold start hours of unit i (h)
T_i^I	Initial status of unit i (h)
$X_{i,t,k}$	Fixed length of the partitions k of the variable $p_{i,t}$ used in the approximation of production costs (MW)
$Y_{i,t,k}$	Fixed length of the partitions k of the variable $\pi_{i,t}$ used in the approximation of production costs (MW^2)

Continuous variables

$cd_{i,t}$	Shut-down cost of unit i in period t (\$)
$cost$	Total cost (\$)
$cp_{i,t}$	Production cost of unit i in period t (\$)
$cu_{i,t}$	Startup cost of unit i in period t (\$)
$p_{i,t}$	Power output of unit i in period t (MW)
$\overline{p}_{i,t}$	Maximum power output of unit i in period t (MW)
$pu_{i,t}$	Partial power output of unit i that is above P_i^L (MW)
$r_{i,t}$	Reserves provided by unit i in period t (MW)
$\gamma_{i,t,k}$	Incremental power output of unit i in period t in the segment k of the partition scheme of $p_{i,t}$ (MW)
$\pi_{i,t}$	Approximation of the quadratic term $p_{i,t}^2$ (MW^2) Binary variables
$u_{i,t}$	On/off status of unit i in period t
$u_{i,t}^{up}$	Startup status of unit i in period t
$u_{i,t}^{dn}$	Shutdown status of unit i in period t
yon_i	On/off toggle status of unit i during the time horizon
$\delta_{i,l,t}$	Indicates the type of startup as a function of the previous shutdown

References

Alemany, J., Magnago, F., Moitre, D., Pinto, H., 2014. Symmetry issues in mixed integer programming based Unit Commitment. Int. J. Elec. Power 54, 86–90.

Arroyo, J.M., Conejo, A.J., 2000. Optimal response of a thermal unit to an electricity spot market. IEEE T. Power Syst. 15, 1098–1104.

Balas, E., 1985. Disjunctive programming and a hierarchy of relaxations for discrete optimization problems. SIAM J. Algebra. Discr. 6, 466–486.

Benhamida, F., Abdelbar, B., 2010. Enhanced Lagrangian relaxation solution to the generation scheduling problem. Int. J. Elec. Power 32, 1099–1105.

Bixby, R., Rothberg, E., 2007. Progress in computational mixed integer programming- a look back from the other side of the tipping point. Ann. Oper. Res. 49, 37–41.

Carrion, M., Arroyo, J.M., 2006. A computationally efficient mixed-integer linear formulation for the thermal unit commitment problem. IEEE T. Power Syst. 21, 1371–1378.

Chang, G., Aganagic, M., Waight, J., Medina, J., Burton, T., Reeves, S., Christoforidis, M., 2001. Experiences with mixed integer linear programming based approaches on short-term hydro scheduling. IEEE T. Power Syst. 16, 743–749.

Dekrajangpetch, S., Sheble, G., Conejo, A., 1999. Auction implementation problems using LaGrangian Relaxation. IEEE T. Power Syst. 14, 82–88.

Dentcheva, D., Gollmer, R., Möller, A., Römisch, W., Schultz, R., 1997. Progress in Industrial Mathematics at ECMI 96. B. G. Teubner GmbH. chapter Solving the Unit Commitment Problem in Power Generation by Primal and Dual Methods.

Dereu, G., Grellier, V., 2010. Handbook of power systems I. Springer, Berlin Heidelberg. chapter Latest improvements of EDF mid-term power generation management. Energy systems, pp. 77–94.

Dolan, E., Moré, J., 2002. Benchmarking optimization software with performance profiles. Math. Program. 91, 201–213.

Fan, W., Guan, X.H., Zhai, Q.Z., 2002. A new method for unit commitment with ramping constraints. Electr. Pow. Syst. Res. 62, 215–224.

Frangioni, A., 2010. Unit commitment problems: A tale in lagrangian optimization. Optima 84.

Frangioni, A., Gentile, C., Lacalandra, F., 2008. Solving unit commitment problems with general ramp constraints. Int. J. Elec. Power 30, 316–326.

Frangioni, A., Gentile, C., Lacalandra, F., 2009. Tighter approximated MILP formulations for unit commitment problems. IEEE T. Power Syst. 24, 105–113.

Friedman, Z., Ingalls, J., Siirola, J.D., Watson, J.P., 2013. Block-oriented modeling of superstructure optimization problems. Comput. Chem. Eng. 57, 10 – 23. PSE-2012.

Hechme-Doukopoulos, G., Brignol-Charousset, S., Malick, J., Lemarchal, C., 2010. The short-term electricity production management problem at EDF. Optima 84.

Hedman, K., Ferris, M., O'Neill, R., Fisher, E., Oren, S., 2010. Co-optimization of generation unit commitment and transmission switching with n-1 reliability. IEEE T. Power Syst. 25, 1052–1063.

Hodge, B.M.S., Huang, S., Siirola, J.D., Pekny, J.F., Reklaitis, G.V., 2011. A multi-paradigm modeling framework for energy systems simulation and analysis. Comput. Chem. Eng. 35, 1725 – 1737.

Kazarlis, S.A., Bakirtzis, A.G., Petridis, V., 1996. A genetic algorithm solution to the unit commitment problem. IEEE T. Power Syst. 11, 83–90.

Koltsaklis, N., Georgiadis, M., 2015. An integrated unit commitment and generation expansion planning model, in: Gernaey, K., Huusom, J., Gani, R. (Eds.), 12th International Symposium on Process Systems Engineering & 25th European Symposium on Computer Aided Process Engineering. Elsevier, pp. 2273–2278.

Lee, J., Leung, J., Margot, F., 2004. Min-up/min-down polytopes. Discrete Optimization 1, 77–85.

Li, T., Shahidehpour, M., 2005. Price-based unit commitment: A case of Lagrangian relaxation versus mixed integer programming. IEEE T. Power Syst. 20, 2015–2025.

Liberti, L., 2012. Reformulations in mathematical programming: automatic symmetry detection and exploitation. Math. Program. 131, 273–304.

Lima, R.M., Grossmann, I.E., 2011. Chemical Engineering Greetings to Prof. Sauro Pierucci. AIDIC. chapter Computational advances in solving Mixed Integer Linear Programming problems. pp. 151–160.

Lima, R.M., Grossmann, I.E., Jiao, Y., 2011. Long-term scheduling of a single-stage multi-product continuous process to manufacture high performance glass. Comput. Chem. Eng. 35, 554–574.

Lima, R.M., Marcovecchio, M., Novais, A., Grossmann, I., 2013. On the computational studies of deterministic global optimization of head dependent short-term hydro scheduling. IEEE T. Power Syst. 28, 4336–4347.

Liu, P., Georgiadis, M.C., Pistikopoulos, E.N., 2013. An energy systems engineering approach for the design and operation of microgrids in residential applications. Chem. Eng. Res. Des. 91, 2054–2069.

Marcovecchio, M.G., Novais, A.Q., Grossmann, I.E., 2011. A deterministic optimization approach for the unit commitment problem, in: Pistikopoulos, EN and Georgiadis, MC and Kokossis, AC (Ed.), 21st European Symposium on Computer Aided Process Engineering, pp. 532–536.

Marcovecchio, M.G., Novais, A.Q., Grossmann, I.E., 2014. Deterministic optimization of the thermal unit commitment problem: A branch and cut search. Comput. Chem. Eng. 67, 53 – 68.

Margot, F., 2010. Symmetry in integer linear programming, in: Jünger, M., Liebling, T.M., Naddef, D., Nemhauser, G.L., Pulleyblank, W.R., Reinelt, G., Rinaldi, G., Wolsey, L.A. (Eds.), 50 Years of Integer Programming 1958-2008. Springer Berlin Heidelberg, pp. 647–686.

Mitra, S., 2013. Optimal multi-scale demand-side management for continuous power-intensive processes. Ph.D. thesis. Carnegie Mellon University.

Mitra, S., Grossmann, I.E., Pinto, J.M., Arora, N., 2012. Optimal production planning under time-sensitive electricity prices for continuous power-intensive processes. Comput. Chem. Eng. 38, 171 – 184.

Mitra, S., Sun, L., Grossmann, I.E., 2013. Optimal scheduling of industrial combined heat and power plants under time-sensitive electricity prices. Energy 54, 194 – 211.

Morales-Espana, G., Latorre, J., Ramos, A., 2013. Tight and compact MILP formulation for the thermal unit commitment problem. IEEE T. Power Syst. 28, 4897–4908.

Nowak, M.P., Romisch, W., 2000. Stochastic Lagrangian relaxation applied to power scheduling in a hydro-thermal system under uncertainty. Ann. Oper. Res. 100, 251–272.

Ostrowski, J., Anjos, M., Vannelli, A., 2014. Modified orbital branching for structured symmetry with an application to unit commitment. Math. Program. , 1–31.

Ostrowski, J., Anjos, M.F., Vannelli, A., 2012. Tight mixed integer linear programming formulations for the unit commitment problem. IEEE T. Power Syst. 27, 39–46.

Padhy, N., 2004. Unit commitment - A bibliographical survey. IEEE T. Power Syst. 19, 1196–1205.

Rajan, D., Takriti, S., 2005. Minimum Up/Down Polytopes of the Unit Commitment Problem With Start-Up Costs. Technical Report. IBM Research.

Redondo, J.N., Conejo, A.J., 1999. Short-term hydro-thermal coordination by Lagrangian relaxation: solution of the dual problem. IEEE T. Power Syst. 14, 89–95.

Rey, P.A., 2004. Eliminating redundant solutions of some symmetric combinatorial integer programs. Electronic Notes in Discrete Mathematics 18, 201 – 206.

Rothberg, E., 2007. An evolutionary algorithm for polishing mixed integer programming solutions. INFORMS J. Comput. 19, 534–541.

Sherali, H., Smith, J., 2001. Improving discrete model representations via symmetry considerations. Manage. Sci. 47, 1396–1407.

Sherali, H.D., 2001. On mixed-integer zero-one representations for separable lower-semicontinuous piecewise-linear functions. Oper. Res. Lett. 28, 155 – 160.

Siirola, J.D., Watson, J.P., 2012. Modeling and Optimization of Superstructure-based Stochastic Programs for Risk-aware Decision Support, in: Karimi, I.A. and Srinivasan, R. (Ed.), 11th International Symposium on Process Systems Engineering, pts A and B, pp. 1060–1064.

Sioshansi, R., O'Neill, R., Oren, S.S., 2008. Economic consequences of alternative solution methods for centralized unit commitment in day-ahead electricity markets. IEEE T. Power Syst. 23, 344–352.

Soroush, M., Chmielewski, D.J., 2013. Process systems opportunities in power generation, storage and distribution. Comput. Chem. Eng. 51, 86 – 95.

Vecchietti, A., Lee, S., Grossmann, I.E., 2003. Modeling of discrete/continuous optimization problems: characterization and formulation of disjunctions and their relaxations. Comput. Chem. Eng. 27, 433 – 448.

Williams, H., 1999. Model building in mathematical programming. Wiley.

Wolsey, L., 1998. Integer Programming. Wiley.

Wood, A.J., Wollenberg, B., 1996. Power generation, operation, and control. New York: Wiley. 2nd Ed. edition.

Xiao, J., Hodge, B.M.S., Pekny, J.F., Reklaitis, G.V., 2011. Operating reserve policies with high wind power penetration. Comput. Chem. Eng. 35, 1876 – 1885.

Zhai, Q., Guan, X., Cui, J., 2002. Unit commitment with identical units: Successive subproblem solving method based on Lagrangian relaxation. IEEE T. Power Syst. 17, 1250–1257.

Zhou, Z., Botterud, A., Wang, J., Bessa, R., Keko, H., Sumaili, J., Miranda, V., 2012. Application of probabilistic wind power forecasting in electricity markets. Wind Energy 16, 321–338.

Zondervan, E., Grossmann, I.E., de Haan, A.B., 2010. Energy optimization in the process industries: Unit commitment at systems level, in: Pierucci, S., Ferraris, G.B. (Eds.), 20th European Symposium on Computer Aided Process Engineering. Elsevier. volume 28, pp. 931 – 936.