A COLUMN GENERATION APPROACH FOR SOLVING VERY LARGE SCALE INSTANCES OF THE BRAZILIAN LONG TERM POWER EXPANSION PLANNING MODEL

Carlos Henrique Medeiros de Sabóia
CEPEL - Electric Power Research Center
COPPE/UFRJ - Systems Engineering Program
Rio de Janeiro, Brazil
saboia@cepel.br

Abilio Lucena
UFRJ - Faculty of Management
COPPE/UFRJ - Systems Engineering Program
Rio de Janeiro, Brazil
abiliolucena@globo.com

Abstract - This paper applies the Dantzig-Wolfe decomposition principle to the mixed integer programming based Brazilian long term power expansion planning model. Since no formulation closely resembling the specific details we have to address could be found in the literature, we had to impose significant modifications to a Dantzig-Wolfe based reformulation and solution algorithm previously suggested for a power planning application. As a result, minimum mandatory thermal power plants operation levels could be imposed and were adequately dealt with within a column generation scheme. The solution algorithm we implemented initially suffered from severe stabilization problems. These were eventually settled through a nonstandard procedure we suggested and the algorithm then proved quite effective. Indeed, it was capable of solving very large scale real world instances of our planning problem, to an optimality guarantee of 1% or less. For some of these instances, such a guarantee is currently out of reach of state-of-the-art commercial optimization software.

Keywords - power generation expansion planning, mixed integer linear programming, dantzig-wolfe decomposition, column generation, stabilization methods.

1 INTRODUCTION

The Brazilian long term power expansion planning model - MELP [1] is used by the Brazilian Government as an official planning tool. An example of its use is the Brazilian National Energy Plan [3]. The generation expansion problem associated with that application, is Mixed Integer Linear Programming (MILP) based. It associates binary 0-1 variables to candidate hydro/thermal power plants and interconnection transmission lines. Additionally, continuous variables are used to describe system operation under, among others, hydrological uncertainties and load curves.

So far, the Mixed Integer Programming module of CPLEX 11.0 has been successfully used as a solution algorithm for MELP. Indeed, CPLEX proved very efficient in solving a number of the test cases found in [3]. However, as the planning horizon is pushed forward and different load curves are used for every seasonal period, the problem becomes out of reach of that software. As an example, take a typical instance spanning a twenty five years planning horizon. It contains a portfolio of approximately one thousand hydro, thermal and interconnection line projects. Additionally, it involves three load curves under seasonal generation conditions. Finally, it requires half a million variables and almost a million constraints. For such an instance, even finding the Linear Programming (LP) relaxation bound proved difficult, due to matrix fill up.

In order to tackle instances such as the one indicated above, we have developed a Dantzig-Wolfe based decomposition algorithm for MELP. Although such an approach is not new to the electric sector [4, 6], some specific details we faced prevented us from directly adapting existing formulations and algorithms to our application. We have thus introduced a tailor made reformulation that has similarities with those suggested for power distribution expansion, in [4]. However, contrary to that reference, our reformulation is capable of accommodating conditions such as those imposed by the minimum mandatory generating levels required by thermal power plants.

Details of our MELP reformulation, the stabilization we carried out for column generation, the solution algorithm we proposed and the computational experiments we performed, are organized as we describe next. In Section 2, we briefly describe MELP and suggest a compact, exposition oriented, formulation for it. Section 3 concentrates on the capacity-planning reformulations proposed in [4] and highlights the main difficulties we have found in directly adapting them to MELP. The MELP reformulation we propose and details related to the application of Dantzig-Wolfe decomposition to it, are presented in sections 4 to 6. Computational experiments for an associated solution algorithm are described in Section 7, where very large scale instances of MELP are solved with an optimality guarantee of 1% or less. Finally, Section 8 closes this paper with some conclusions and a discussion of the future work we intend to do.

2 THE BRAZILIAN MODEL

The power expansion planning problem addressed in [1] is formulated as a large scale MILP problem where electric and natural gas (NG) systems are modeled in an integrated pattern [2]. For the electric system, over the suggested planning horizon, investment on hydro/thermal power plants and interconnection transmission lines is modeled through binary 0-1 variables. Additionally, for every planning horizon time, the operation problem for this system is modeled as a demand satisfaction problem, involving continuous variables. For such a problem, the energy demanded is attempted to be met through available candidate power plants and interconnection flow lines. Taking into account peak and off-peak load curves, demand satisfaction is enforced for average and critical hydrological conditions. Conveniently written coupling constraints are used to link project capacities to operation variables.

For the NG system, investment on pipelines and liquefied natural gas (LNG) regasification facilities is also modeled through binary 0-1 variables. On the other hand, the operation problem to be addressed for this system is to determine the amount of natural gas required either inside or outside the overall electric/NG system. Within the NG system, natural gas is required for the operation of some of the available NG thermal power plants. For the Brazilian electric power system, it is reasonable to assume that thermal power plants powered by fuels other than gas are not subject to supply shortages.

Prior to introducing a presentation oriented compact formulation of the problem in [1, 2], let us consider the following associated data:

- $J = J_H \cup J_T \cup J_I \cup J_G$ is the set of available projects. It subdivides into hydro (J_H) and thermal (J_T) power plants as well as interconnection transmission lines (J_I) and pipelines and LNG regasifying facilities (J_G). For simplicity, we denote by $J_E = J \setminus J_G$ the overall electric system;
- $\{x_i^k \in \{0,1\} : i \in J\}$ are investment decision variables for those projects available at time k;
- Constant ϕ_i, for $i \in J$, is the investment cost for project i. A corresponding discounted value $\phi_i^k = \frac{\phi_i}{(1+\tau)^k}$, at discount rate τ, is also available for every time k of the planning horizon.
- Variables $\{y_{i,l}^{cr,k} \in \mathbb{R}_+ : i \in J\}$ and $\{y_{i,l}^{av,k} \in \mathbb{R}_+ : i \in J\}$ are respectively defined for critical and average hydrological conditions at time k, under load curve l. They express four different entities: power generation volumes at hydro/thermal power plants, volumes of imported LNG, and net flows for interconnection transmission lines and NG pipelines. Additionally, they also express energy deficits associated with the electric and the NG systems. Under the critical condition, energy deficits are required to be null. Under the average condition, at a cost, they are allowed to be nonzero;
- Under average hydrological conditions, constant γ_i, for $i \in J$ gives, whatever applies, thermal generation cost or deficit level cost. Additionally, depending on what i represents, γ_i may also give pipeline operation cost or LNG import cost. As before, $\gamma_i^k = \frac{\phi_i}{(1+\tau)^k}$ gives, under a discount rate τ, the corresponding discounted value of γ_i at time k;
- For a given time period k and load curve l, constants $\{G_{i,l}^k : i \in J\}$ give minimum required generation and flow levels.
- For a given time period k and load curve l, constants $\{H_{i,l}^k : i \in J\}$ give maximum generation and flow levels, whatever applies to i.
- Associated with hydro and thermal power plants at time period k, constants $\{E_i^{cr,k} : i \in J\}$ and $\{E_i^{av,k} : i \in J\}$ respectively give critic and average energy levels under their corresponding hydrological conditions.
- For the set $\mathcal{E}$ of available electric subsystems, Ψ_e identifies those projects belonging to subsystem $e \in \mathcal{E}$. Thus, for the overall electric system, available projects are identified by $J_E = \bigcup_{\forall e \in \mathcal{E}} \Psi_e$;
- Constant $D_l^{e,k}$ defines the amount of energy demanded by subsystem $e \in \mathcal{E}$, at time k, under load curve l;
- For the set $\mathcal{G}$ of available NG subsystems, Γ_g identifies those projects belonging to subsystem $g \in \mathcal{G}$. Thus, for the overall NG system, available projects are given by $J_G = \bigcup_{\forall g \in \mathcal{G}} \Gamma_g$;
- Constant $D^{g,k}$ defines the amount of nonelectric energy demanded by subsystem $g \in \mathcal{G}$, at time k;
- J_T^g defines the set of NG thermal power plants that are geographically located within NG subsystem g. These plants satisfy the relation $\bigcup_{g \in \mathcal{G}} J_T^g \subseteq J_T$;
- Index sets $\mathcal{P}$ and $\mathcal{L}$ are respectively associated with the planning horizon and load curves;
- ρ is a conversion factor that transforms electric energy unit into volume of natural gas;

To further simplify the notation, we will generically denote by $y_{i,l}^{p,k}$, where $p \in \{cr, av\}$, any of the previously defined $\mathbf{y}$ variables. As such the generation expansion problem addressed here is formulated as

$$\min \quad \sum_{k \in \mathcal{P}} \left(\sum_{i \in J} \phi_i^k x_i^k + \sum_{i \in J_{T \cup G}} \sum_{l \in L} \gamma_i^k y_{i,l}^{av,k} \right) \quad (1)$$

$$\text{s.t.} \quad \sum_{k \in \mathcal{P}} x_i^k \leq 1, \ \forall i \in J \quad (2)$$

$$\sum_{i \in \Psi_e} y_{i,l}^{p,k} = D_l^{e,k}, \ \forall (e,k,l) \in \{\mathcal{E}, \mathcal{P}, \mathcal{L}\} \quad (3)$$

$$\sum_{\substack{i \in \Gamma_g, \\ l \in \mathcal{L}}} y_{i,l}^{p,k} = D^{g,k} + \sum_{\substack{i \in J_T^g, \\ l \in \mathcal{L}}} \rho \cdot y_{i,l}^{p,k}, \ \forall (g,k) \in \{\mathcal{G}, \mathcal{P}\} \quad (4)$$

$$\sum_{l \in L} y_{i,l}^{p,k} \leq \sum_{n=1,\dots,k} E_i^{p,k} x_i^n, \ \forall (i,k) \in \{J_{H \cup T}, \mathcal{P}\} \quad (5)$$

$$y_{i,l}^{p,k} \leq \sum_{n=1,\dots,k} H_{i,l}^k x_i^n, \ \forall (i,k,l) \in \{J, \mathcal{P}, \mathcal{L}\} \quad (6)$$

$$y_{i,l}^{p,k} \geq \sum_{n=1,\ldots,k} G_{i,l}^{k} x_i^n, \ \begin{smallmatrix}\forall (i,k,l) \in \\ \{J,\mathcal{P},\mathcal{L}\}\end{smallmatrix} \quad (7)$$

$$x_i^k \in \{0,1\}, \ \forall (i,k) \in \{J,\mathcal{P}\} \quad (8)$$

$$y_{i,l}^{p,k} \geq 0, \ \forall (i,k,l) \in \{J,\mathcal{P},\mathcal{L}\} \quad (9)$$

In this formulation, investment constraints are modeled through knapsack type inequalities (2). Constraints (3)-(4) represent the electric energy and NG demand satisfaction levels. One should notice that these equations are coupled through the NG thermal power plants generation. Finally, the set of constraints (5)-(7) links the investment and operation problems through generation limits.

Additionally, in the most recent version of MELP, the operation problem may be analyzed in a seasonal basis. From the computational point of view, the continuous variables $\mathbf{y}$, as well as the operation and linking constraints are multiplied by the number of seasons in order to allow this representation.

3 REFORMULATIONS FROM THE LITERATURE

Until quite recently, solver CPLEX 11.0 was used as a solution algorithm for (2)-(9). However, for large scale instances of our planning problem, such as the one we previously mentioned, CPLEX proved unable to find good quality feasible solutions in acceptable CPU times. Indeed, for a 2.83Ghz Intel Core 2Quad Q9550 based machine, under 4Gb of RAM, even solving the Linear Programming (LP) relaxation of (2)-(9), either by the Simplex Method or an Interior Point Algorithm, proved difficult (matrix fill up being the reported reason).

A quite natural way of attempting to overcome the difficulties raised above is to resort to the use of decomposition methods. For our application, we have opted for using Dantzig-Wolfe decomposition. The reason being the impressive results obtained with this method for a wide range of applications. An additional reason is the promising results recently obtained for power planning in [4].

A straightforward Dantzig-Wolfe decomposition of (2)-(9) is obtained by noticing that constraints (5)-(7) couple the $|\mathcal{P}|$ continuous structures defined by (3) and (4) with the knapsack problem defined by (2). Therefore one could build a Master Problem (MP) over (5)-(7) and thus have independent Subproblems (SPs) being implied by (2)-(4). However, this approach would lead to a MP without a clear structure to exploit. Furthermore, within SPs, no coupling would exist between investment related integer variables and the continuous variables used for system operation. Therefore columns generated for MP would bring in limited information.

Due to the limitations above, we decided to investigate the use of the reformulations proposed in [4]. These were originally devised for tackling multi-stage stochastic capacity planning problems. However, for our application, they were limited to a single, thus deterministic, scenario.

The decomposition scheme suggested in [4] is based on *variable splitting*, a popular Stochastic Programming technique. For more details see [5] that applied the concept to, among others, plant location problems. It is carried out after introducing variables $\{x_i^{hk} \in \{0,1\} : i \in J, \ (h,k) \in \mathcal{P}, \ k \geq h\}$ into a formulation somewhat like (2)-(6). Accordingly, $x_i^{hk} = 1$ ensures that project i is operational at time k since it first became operational at time h. Leaving inequalities (7) aside, formulation (2)-(9) could thus be reformulated, under these new variables, as

$$\min \quad \sum_{k \in \mathcal{P}} \left(\sum_{i \in J} \phi_i^k x_i^k + \sum_{i \in J_{T \cup G}} \sum_{l \in L} \gamma_i^k y_{i,l}^{av,k} \right)$$

$$\text{s.t.} \quad \sum_{h \in \mathcal{P}} x_i^h \leq 1, \ \forall i \in J \quad (10)$$

$$x_i^{hk} \leq x_i^h, \ \forall i \in J, \ \forall (h,k) \in \mathcal{P}, \ k \geq h \quad (11)$$

$$\sum_{i \in \Psi_e} y_{i,l}^{p,k} = D_l^{e,k}, \ \begin{smallmatrix}\forall (e,k,l) \in \\ \{\mathcal{E},\mathcal{P},\mathcal{L}\}\end{smallmatrix} \quad (12)$$

$$\sum_{\substack{i \in \Gamma_g, \\ l \in \mathcal{L}}} y_{i,l}^{p,k} = D^{g,k} + \sum_{\substack{i \in J_T^g, \\ l \in \mathcal{L}}} \rho \cdot y_{i,l}^{p,k}, \ \begin{smallmatrix}\forall (g,k) \in \\ \{\mathcal{G},\mathcal{P}\}\end{smallmatrix} \quad (13)$$

$$\sum_{l \in L} y_{i,l}^{p,k} \leq \sum_{h=1,\ldots,k} E_i^{p,k} x_i^{hk}, \ \begin{smallmatrix}\forall (i,k) \in \\ \{J_{H \cup T},\mathcal{P}\}\end{smallmatrix} \quad (14)$$

$$y_{i,l}^{p,k} \leq \sum_{h=1,\ldots,k} H_{i,l}^{k} x_i^{hk}, \ \begin{smallmatrix}\forall (i,k,l) \in \\ \{J,\mathcal{P},\mathcal{L}\}\end{smallmatrix} \quad (15)$$

$$x_i^k \in \{0,1\}, \ \forall (i,k) \in \{J,\mathcal{P}\} \quad (16)$$

$$x_i^{hk} \in \{0,1\}, \ \forall i \in J, \ \forall (h,k) \in \mathcal{P}, \ k \geq h \quad (17)$$

$$y_{i,l}^{p,k} \geq 0, \ \forall (i,k,l) \in \{J,\mathcal{P},\mathcal{L}\} \quad (18)$$

Reformulation (10)-(18) presents two main advantages over (2)-(9). The first one is that quite a bit of duplication of information exists in it. As a result, a Dantzig-Wolf decomposition could be devised where MP and SPs are more closely connected. This is accomplished, for instance, by basing MP on constraints (10) and (11) while SPs on (12)-(15). In doing so, notice that every SP now involves not only continuous variables but binary ones as well. Furthermore notice that integer variables in MP share a lot of information with integer variables in SPs. As a result, as column generation proceeds, these two types of problems should greatly benefit from the exchange of information being induced by the decomposition. The generation of better quality columns should thus be expected.

As for the second advantage, notice that the number of column entries is now significantly smaller than before, as a very large number of constraints would have been moved into SPs. Furthermore, integer and continuous variables are now tightly coupled within SPs, thus generating more attractive columns for MP. Finally, MP now possess a structure that will allow feasible LP relaxation solutions to be generated for it.

In addition to the reformulation above, a more compact reformulation of (2)-(6), that does not use split variables $\{x_i^{hk} \in \{0,1\} : i \in J, \ (h,k) \in \mathcal{P}, \ k \geq h\}$, has also been suggested in [4]. It appears more convenient to use than the previous one since it leads to more compacts MP and SPs. The reformulation uses a new set of variables $\{z_i^k \in \{0,1\} : i \in J,\ k \in \mathcal{P}\}$ and is given by

$$\min \quad \sum_{k\in\mathcal{P}} \left(\sum_{i\in J} \phi_i^k x_i^k + \sum_{i\in J_{T\cup G}} \sum_{l\in L} \gamma_i^k y_{i,l}^{av,k} \right)$$

$$\text{s.t.} \quad \sum_{h\in\mathcal{P}} x_i^h \le 1,\ \forall i \in J \qquad (19)$$

$$z_i^k \le \sum_{h=1,\ldots,k} x_i^h,\ \forall i \in J,\ \forall k \in \mathcal{P} \qquad (20)$$

$$\sum_{i\in\Psi_e} y_{i,l}^{p,k} = D_l^{e,k},\ \begin{smallmatrix}\forall(e,k,l)\in\\ \{\mathcal{E},\mathcal{P},\mathcal{L}\}\end{smallmatrix} \qquad (21)$$

$$\sum_{\substack{i\in\Gamma_g,\\ l\in\mathcal{L}}} y_{i,l}^{p,k} = D^{g,k} + \sum_{\substack{i\in J_T^g,\\ l\in\mathcal{L}}} \rho \cdot y_{i,l}^{p,k},\ \begin{smallmatrix}\forall(g,k)\in\\ \{\mathcal{G},\mathcal{P}\}\end{smallmatrix} \qquad (22)$$

$$\sum_{l\in L} y_{i,l}^{p,k} \le E_i^{p,k} z_i^k,\ \begin{smallmatrix}\forall(i,k)\in\\ \{J_{H\cup T},\mathcal{P}\}\end{smallmatrix} \qquad (23)$$

$$y_{i,l}^{p,k} \le H_{i,l}^k z_i^k,\ \begin{smallmatrix}\forall(i,k,l)\in\\ \{J,\mathcal{P},\mathcal{L}\}\end{smallmatrix} \qquad (24)$$

$$x_i^k \in \{0,1\},\ z_i^k \in \{0,1\},\ \forall(i,k) \in \{J,\mathcal{P}\} \qquad (25)$$

$$y_{i,l}^{p,k} \ge 0,\ \forall(i,k,l) \in \{J,\mathcal{P},\mathcal{L}\} \qquad (26)$$

Despite the attractive features found in the two reformulations we have just described, none of them is directly capable of dealing with an important feature of the problem we face. Namely, they do not directly accommodate minimum mandatory operation levels imposed on thermal power plants.

Indeed, various practical real world energy planning applications exist where the use of inequalities such as (7) must be enforced. For example, take the Brazilian case, where, due to established fuel contracts or operation requirements, some thermal power plants, like coal fired or nuclear based ones, must operate as just suggested. Additionally, one might also have to comply with pre-defined minimum import levels of NG, to justify the investments made on the distribution network for this commodity.

Noticing that nonzero costs are associated with the operation of the thermal power plants, let us adapt inequalities (7) to be used in the two previous reformulations. Accordingly, for conveniently defined indices, inequalities $\left(y_{i,l}^{p,k} \ge \sum_{h=1,\ldots,k} G_{i,l}^k x_i^{hk}\right)$ would result for the first reformulation while $\left(y_{i,l}^{p,k} \ge G_{i,l}^k z_i^k\right)$ would apply for the second. However, under these two sets of inequalities, at time k, the nonzero cost thermal power plant $\mathbf{y}$ variables would drive corresponding variables x_i^{hk} and z_i^k to zero, unless energy demand push them, otherwise, towards 1. This should be the case even when i becomes operational prior to k. In other words, simply bringing these inequalities into their corresponding reformulations would not ensure that minimum mandatory generation levels are observed. A way out of this situation is to turn inequalities (11) and (20) into corresponding equality constraints, i.e.,

$$x_i^{hk} = x_i^h,\ \forall i \in J,\ \forall(h,k) \in \mathcal{P},\ k \ge h \qquad (27)$$

$$z_i^k = \sum_{h=1,\ldots,k} x_i^h,\ \forall i \in J,\ \forall k \in \mathcal{P} \qquad (28)$$

We have thus borrowed some ideas from the two reformulations in [4] to introduce an extended reformulation that adequately deals with the conditions imposed on the operation of thermal power plants.

4 AN EXTENDED REFORMULATION

For our application, for reasons that will become evident later on, a MELP reformulation suitable for column generation should ideally contemplate:

1. the incorporation of constraints (5)-(7) into SPs, thus reducing MP scale;
2. the formulation of SPs that are not polynomial time problems but are still *simple enough* to be solved to proven integral optimality (thus allowing *good quality* columns to be generated for MP);
3. the formulation of a MP that is capable of dealing with the infeasible sets of columns eventually generated at SPs, i.e., an MP where variables corresponding to these columns would naturally assume a null value, as the algorithm converges to optimum;
4. have enough structure and sparsity to allow the LP solver to obtain LP relaxation solutions;

For $\pi = |\mathcal{P}|$, let $\delta_h = \pi - h + 1,\ \forall h \in \mathcal{P}$, be the number of time periods spanned from h till the end of the planning horizon, i.e., time $|\mathcal{P}|$. As we explain next, an alternative path to accommodate constraints (7) into reformulation (10)-(18) is based on constraints

$$\sum_{k=h}^{\pi} x_i^{hk} = \delta_h x_i^h,\ \forall i \in J,\ \forall h \in \mathcal{P}. \qquad (29)$$

Proposition 1 *Feasible solutions to (11),(29) and to (27), where the variables involved are as previously defined, are in a one-to-one correspondence.*

Proof 1 *Let $(\bar{x}_i^h, \bar{x}_i^{hk})$ be a feasible solution to (11),(29). Then, due to the upper limit (x_i^h) imposed on variables (x_i^{hk}) by constraints (11), and the equality sign of constraint (29), $(\bar{x}_i^h, \bar{x}_i^{hk})$ must also be feasible for (27). Conversely, if $(\hat{x}_i^h, \hat{x}_i^{hk})$ is feasible for (27) and one takes the sum of these constraints over k, one arrives at $\sum_{k=h}^{\pi} x_i^{hk} = \sum_{k=h}^{\pi} x_i^h = \delta_h x_i^h,\ \forall i \in J,\ \forall h \in \mathcal{P}$. Therefore $(\hat{x}_i^h, \hat{x}_i^{hk})$ is feasible to (11),(29).*

As we justify next, one could make reformulation (10)-(18), (29) more compact by replacing constraints (11) with surrogate constraint

$$\sum_{h=1}^{k} x_i^{hk} \le \sum_{h=1}^{k} x_i^h,\ \forall i \in J,\ \forall k \in \mathcal{P} \qquad (30)$$

Proposition 2 *Under the binary* 0-1 *variables involved, the feasibility region defined by the intersection of constraints (29) and (30) is the same as that defined by the intersection of the first set of constraints with an equality constrained version of the second.*

Proof 2 *Take* $\{\sum_{h=1}^{\pi}\sum_{k=h}^{\pi} x_i^{hk} = \sum_{h=1}^{\pi}\delta_h \cdot x_i^h,\ \forall i \in J\}$ *as a surrogate of (29) and compare it with a surrogate* $\{\sum_{k=1}^{\pi}\sum_{h=1}^{k} x_i^{hk} \le \sum_{k=1}^{\pi}\sum_{h=1}^{k} x_i^h,\ \forall i \in J\}$, *for (30). After some algebraic manipulation, the surrogate of (29) could be written as* $\{\sum_{k=1}^{\pi}\sum_{h=1}^{k} x_i^{hk} = \sum_{k=1}^{\pi}\sum_{h=1}^{k} x_i^h,\ \forall i \in J\}$. *and the two sets of constraints thus have the same left and right hand-sides. As such, the result thus follows.*

Finally, to reduce the large number of binary 0-1 SP variables and also to adequately deal with the infeasible columns SPs may eventually return to MP, variables $\{z_i^k \in \{0,1\} : i \in J,\ k \in \mathcal{P}\}$ and $\{w_i^h \in [0,1] : i \in J,\ h \in \mathcal{P}\}$ are introduced into our reformulation. Additionally, variables $\left\{x_i^{hk} : i \in J, (h,k) \in \mathcal{P}\right\}$, that were previously defined as binary 0-1, are now defined over the $[0,1]$ interval. The reasoning behind this decision is given in the next section. Our resulting MELP reformulation is given by

$$\min \quad \sum_{k\in\mathcal{P}}\left(\sum_{i\in J}\phi_i^k x_i^k + \sum_{i\in J_{T\cup G}}\sum_{l\in L}\gamma_i^k y_{i,l}^{av,k}\right)$$

$$\text{s.t.} \quad \sum_{h\in\mathcal{P}} x_i^h \le 1,\ \forall i \in J \quad (31)$$

$$\sum_{k=h}^{|\mathcal{P}|} x_i^{hk} = \delta_h(x_i^h - w_i^h),\ \forall i \in J,\ \forall h \in \mathcal{P} \quad (32)$$

$$\sum_{h=1}^{k} x_i^{hk} \le \sum_{h=1}^{k} x_i^h,\ \forall i \in J,\ \forall k \in \mathcal{P} \quad (33)$$

$$\sum_{h=1}^{k} x_i^{hk} = z_i^k,\ \forall i \in J,\ \forall k \in \mathcal{P} \quad (34)$$

$$\sum_{i\in\Psi_e} y_{i,l}^{p,k} = D_l^{e,k},\ \ \forall (e,k,l)\in\{\mathcal{E},\mathcal{P},\mathcal{L}\} \quad (35)$$

$$\sum_{i\in\Gamma_g,\ l\in\mathcal{L}} y_{i,l}^{p,k} = D^{g,k} + \sum_{i\in J_T^g,\ l\in\mathcal{L}} \rho\cdot y_{i,l}^{p,k},\ \ \forall (g,k)\in\{\mathcal{G},\mathcal{P}\} \quad (36)$$

$$\sum_{l\in L} y_{i,l}^{p,k} \le E_i^{p,k} z_i^k,\ \ \forall (i,k)\in\{J_{H\cup T},\mathcal{P}\} \quad (37)$$

$$y_{i,l}^{p,k} \le H_{i,l}^k z_i^k,\ \ \forall (i,k,l)\in\{J,\mathcal{P},\mathcal{L}\} \quad (38)$$

$$y_{i,l}^{p,k} \ge G_{i,l}^k z_i^k,\ \ \forall (i,k,l)\in\{J,\mathcal{P},\mathcal{L}\} \quad (39)$$

$$x_i^h \in \{0,1\},\ w_i^h \in [0,1],\ \forall i \in J,\ \forall h \in \mathcal{P} \quad (40)$$

$$z_i^k \in \{0,1\},\ \forall i \in J,\ \forall k \in \mathcal{P} \quad (41)$$

$$x_i^{hk} \in [0,1],\ \forall i \in J,\ \forall (h,k) \in \mathcal{P},\ k \ge h \quad (42)$$

$$y_{i,l}^{p,k} \ge 0, \forall i \in J, \forall k \in \mathcal{P}, \forall l \in \mathcal{L} \quad (43)$$

Since subproblems are essentially independent from each other, one should notice that they might generate expansion plans that do not necessarily fit each other. As one may appreciate, the proposed reformulation feasibly accommodates solutions that, taken together, would be infeasible for the whole problem. This is accomplished through the use of slack variables $\mathbf{w}$ that will naturally become zero valued as the decomposition procedure evolves. Furthermore this is attained without having to apply any penalties to those variables, as indicated in the very simple example presented in subsection 5.3.

Differently from here if the reformulations in [4] were used, after incorporating respectively (27) or (28), big-M penalties would have to be applied to the artificial variables associated with these constraints.

5 DANTZIG-WOLFE DECOMPOSITION

Decomposition of (31)-(43) could be conducted through constraints (34) and variables $\mathbf{z}$. To do so, let $\mathbf{Z}_k$ define, at time $k \in \mathcal{P}$, the feasibility set associated with variables $\mathbf{z}$ and $\mathbf{y}$. Accordingly, denote by $\mathbf{z}^k$ and $\mathbf{y}^k$ those variables, respectively in $\mathbf{z}$ and $\mathbf{y}$, corresponding to time k. One could then write $\mathbf{Z}_k = \left\{\mathbf{z}^k : (\mathbf{z}^k, \mathbf{y}^k) \in (\mathbb{B}^{n_1} \times \mathbb{R}_+^{n2}) \cap \{(35)-(39)\}\right\}$, where n_1 and n_2 are conformable dimensions, and $\mathbb{B}^{n_1}$ stands for the n_1-dimensional binary 0-1 space. It should be noticed that, for any $\mathbf{z}^k \in \mathbf{Z}_k$, one will only be interested in operating the system at the least possible cost. Therefore, one should not be bothered, at this stage, with the corresponding $\mathbf{y}^k$ portion of the solution. As we shall see, $\mathbf{y}^k$ will be readily available from SP solutions.

Take $\mathcal{Q}_k$ as the index set for the finite number of points contained in $\mathbf{Z}_k$ and let $\hat{\mathbf{z}}_j^k,\ j \in \mathcal{Q}_k$, denote one such point. Consequently, one may alternatively write $\mathbf{Z}_k = \left\{\hat{\mathbf{z}}_j^k : j \in \mathcal{Q}_k\right\}$. Indeed, for our purposes, a more adequate representation of $\mathbf{Z}_k$ is obtained by restricting attention to the extreme points of that set, indexed by $\hat{\mathcal{Q}}_k \subseteq \mathcal{Q}_k$. Any $\mathbf{z}^k \in \mathbf{Z}_k$ could thus be written as $\mathbf{z}^k = \sum_{j\in\hat{\mathcal{Q}}_k}\lambda_j\hat{\mathbf{z}}_j^k$, where $\sum_{\forall j\in\hat{\mathcal{Q}}_k}\lambda_j = 1,\ \lambda_j \in [0,1],\ \forall j \in \hat{\mathcal{Q}}_k$.

Dropping our matrix notation in favor of an extended one, $(\mathbf{z}^k, \mathbf{y}^k) = (z_i^k, y_{i,l}^{av,k}, y_{i,l}^{cr,k})_{\forall i,l}$ may be written. Likewise, $(\hat{\mathbf{z}}_j^k, \hat{\mathbf{y}}_j^k) = (\hat{z}_i^k, \hat{y}_{i,l}^{av,k}, \hat{y}_{i,l}^{cr,k})_{\forall i,l}^j$ also applies. A description of MP and their corresponding SPs, follows.

5.1 The Master Problem

The MP outlined above is given by

$$\min \quad \sum_{k\in\mathcal{P}}\left(\sum_{i\in J}\phi_i^k x_i^k + \sum_{j\in\hat{\mathcal{Q}}_k}\lambda_j(\gamma_i^k\cdot\hat{y}_{i,l}^{av,k})_{\forall i,l}^j\right)$$

$$\text{s.t.} \quad \sum_{h\in\mathcal{P}} x_i^h \le 1,\ \forall i \in J$$

$$\sum_{k=h}^{|\mathcal{P}|} x_i^{hk} = \delta_h(x_i^h - w_i^h),\ \forall i \in J,\ \forall h \in \mathcal{P}$$

$$\sum_{h=1}^{k} x_i^{hk} \le \sum_{h=1}^{k} x_i^h,\ \forall i \in J,\ \forall k \in \mathcal{P}$$

$$\sum_{h=1}^{k} x_i^{hk} = \sum_{j \in \hat{\mathcal{Q}}_k} \lambda_j (\hat{z}_i^k)^j,\ \forall i \in J,\ \forall k \in \mathcal{P}$$

$$\sum_{j \in \hat{\mathcal{Q}}_k} \lambda_j = 1,\ \forall k \in \mathcal{P}$$

$$x_i^h \in \{0,1\},\ w_i^h \in [0,1],\ \forall i \in J,\ \forall h \in \mathcal{P}$$

$$x_i^{hk} \in [0,1],\ \forall i \in J,\ \forall (h,k) \in \mathcal{P},\ k \geq h$$

$$\lambda_j \in [0,1],\ \forall j \in \hat{\mathcal{Q}}_k,\ \forall k \in \mathcal{P}$$

The number of extreme points of $\mathbf{Z}_k$ is typically very large. Therefore, algorithmically, one should settled for working with a restricted subset of them, $\left\{\hat{\mathbf{z}}_j^k : j \in \hat{\mathcal{Q}}_k'\right\}$, where $\hat{\mathcal{Q}}_k' \subseteq \hat{\mathcal{Q}}_k$. Additional extreme points are to be added to that set, on demand, as they become necessary. If, for the MP formulation above, one substitutes $\hat{\mathcal{Q}}_k$ for $\hat{\mathcal{Q}}_k'$, a Restricted MP (RMP) is formulated instead. Initially, $\hat{\mathcal{Q}}_k' = \{\emptyset\}$, being expanded with columns that correspond to solutions to the SPs we define next. Alternatively, one may warm start the algorithm by initializing it with a few, heuristically generated, extreme points of $\mathbf{Z}_k$.

5.2 The Column Generation Subproblems

At any given iteration j of the decomposition algorithm, after solving the LP relaxation of RMP, dual values corresponding to coupling and convexfication constraints are obtained. For $k \in \mathcal{P}$, respectively denote these values by $\{(\hat{\pi}_i^k)^j : i \in J\}$ and $\{(\hat{\mu}^k)^j\}$.

Let us now concentrate on those equations of (31)-(43) related to system operation. After dualizing coupling constraints (34), $|\mathcal{P}|$ independent subproblems result. Accordingly, at iteration j of the decomposition algorithm, denote by $SP(k)$ the subproblem corresponding to time period $k \in \mathcal{P}$. A formulation for $SP(k)$ is thus given by

$$\min \quad \sum_{i \in J_{T\cup G}} \sum_{l \in L} \gamma_i^k y_{i,l}^{av,k} - \sum_{i \in J} (\hat{\pi}_i^k)^j z_i^k - (\hat{\mu}^k)^j$$

$$\text{s.t.} \quad \sum_{i \in \Psi_e} y_{i,l}^{p,k} = D_l^{e,k},\ \forall (e,l) \in \{\mathcal{E}, \mathcal{L}\}$$

$$\sum_{\substack{i \in \Gamma_g, \\ l \in \mathcal{L}}} y_{i,l}^{p,k} = D^{g,k} + \sum_{\substack{i \in J_T^g, \\ l \in \mathcal{L}}} \rho \cdot y_{i,l}^{p,k},\ \forall g \in \mathcal{G}$$

$$\sum_{l \in L} y_{i,l}^{p,k} \leq E_i^{p,k} z_i^k,\ \forall i \in J_{H\cup T}$$

$$y_{i,l}^{p,k} \leq H_{i,l}^k z_i^k,\ \forall (i,l) \in \{J, \mathcal{L}\}$$

$$y_{i,l}^{p,k} \geq G_{i,l}^k z_i^k,\ \forall (i,l) \in \{J, \mathcal{L}\}$$

$$z_i^k \in \{0,1\},\ \forall i \in J$$

$$y_{i,l}^{p,k} \geq 0, \forall i \in J, \forall l \in \mathcal{L}$$

Provided an optimal solution $(\hat{z}_i^k, \hat{y}_{i,l}^{av,k}, \hat{y}_{i,l}^{cr,k})_{\forall i,l}^j$ to $SP(k)$ has a negative value, it is advantageous to introduce it into RMP. This is accomplished by setting $\hat{\mathcal{Q}}_k' := \hat{\mathcal{Q}}_k' \cup \{j\}$ and, in doing so, it may be possible to further reduce RMP LP relaxation value. Otherwise, if, for any $k \in \mathcal{P}$, no attractive column is generated, the current RMP solution is optimal for MP and, consequently, is optimal for the LP relaxation of (31)-(43). If, additionally, it is integral valued, it would be optimal for (31)-(43).

Whenever, for every iteration of the decomposition algorithm, $\{SP(k) : k \in \mathcal{P}\}$ are solved to proven integral optimality, an optimal solution to MP, and consequently an optimal solution to (31)-(43), lies in between the LP relaxation value of (31)-(43) and the value of an optimal integral solution to the current RMP.

5.3 Justification for the use of variables $\mathbf{z}$ *and* $\mathbf{w}$

Variables $\{z_i^k \in \{0,1\} : i \in J,\ k \in \mathcal{P}\}$ and $\{w_i^h \in [0,1] : i \in J,\ h \in \mathcal{P}\}$ are used in (31)-(43) to accommodate columns that, under previous reformulations, would be infeasible. Additionally, their use help reduce the number of integer variables one would have in SPs.

To illustrate the advantages of using $\mathbf{z}$ and $\mathbf{w}$, let us consider a very small instance of our planning problem, where $|\mathcal{P}|$ = 4 and just one candidate thermal power plant project is available.

Assume that $\hat{z}_1^1 = 1$ and $\hat{z}_1^2 = \hat{z}_1^3 = \hat{z}_1^4 = 0$ is the very first solution returned by SPs. It says that our single project became operational at period 1 but remained inactive from period 2 onwards. It is thus infeasible for MELP since it does not enforce minimum mandatory generation levels for thermal power plants.

For the solution above, constraints (34), used within RMP, ensure that $x_1^{11} = 1$ and $x_1^{12} = x_1^{22} = x_1^{13} = x_1^{23} = x_1^{33} = x_1^{14} = x_1^{24} = x_1^{34} = x_1^{44} = 0$ is obtained. However, from constraints (29), $x_1^1 = 0.25$ and $x_1^2 = x_1^3 = x_1^4 = 0.0$ would result and constraint (30) would thus be violated, since $x_1^{11} > x_1^1$ would be implied.

If one uses (32) instead of (29), $x_1^1 = 1$ and $w_1^1 = 0.75$ would otherwise be obtained. Such a solution is feasible for (32), since $4(1.00 - 0.75) = 1$ results. Furthermore, $x_1^{11} \leq x_1^1$ would hold and (30) would also be satisfied.

In summary, the use of variables $\mathbf{z}$ and $\mathbf{w}$ allows RMP to accommodate infeasible columns eventually returned by SPs. However, this is done at a very high investment cost, which is the ideal compromise, in this case. Therefore, as the column generation procedure advances, variables $\mathbf{w}$ would be driven to zero.

Finally, we should address our use of variables $\mathbf{z}$ which were borrowed from reformulation (19)-(26). Notice that if we were not to use them, continuous variable $x_i^{hk} \in [0,1]$, would have to be otherwise defined as binary 0-1. Furthermore, their solution values would have to be determined at SPs.

6 THE COLUMN GENERATION SCHEME

Our implementation of the Dantzig-Wolfe decomposition principle essentially consists, at any given iteration, in solving the LP relaxation of RMP, formulating SPs with the dual values thus obtained, attempting to solve these SPs to integral optimality with CPLEX 11.0, and expanding RMP with the columns they return. As it is not uncommon for such an scheme, we were faced with severe *column generation stabilization problems.* Although various procedures were suggested in the literature to address this issue the direct use of any of them failed in our case.

Interior Point Algorithm (IPA) based RMP LP relaxation dual values are used in [4], in an attempt to stabilize column generation. However, contrary to [4], the benefits observed for our application were fairly marginal. Eventually, what ended up paying off was to combine such a procedure with the best initialization possible for the stabilization procedure suggested in [7]. Such an initialization requires the LP relaxation of MELP reformulation (31)-(43) to be solved with an IPA. Notice that the dual values thus obtained for the coupling constraints are as good as they could possibly be, if columns are generated through LP relaxations to SPs.

Following the procedure in [7], under our proposed initialization, a sequence of dual values are computed according to convex combinations $(\pi_i^k, \mu^k)^{j+1} = \alpha(\pi_i^k, \mu^k)^{best} + (1-\alpha)(\hat{\pi}_i^k, \hat{\mu}^k)^j$, $\forall i \in J$ and $\forall k \in \mathcal{P}$, where $0 \leq \alpha \leq 1$. Accordingly, $(\pi_i^k, \mu^k)^{best}$ identifies the dual values associated with the best lower bound so far attained for (31)-(43) and is initialized as described above.

An important characteristic of subproblem $SP(k)$, for $k \in \mathcal{P}$, is that it does possess an integrality gap. We therefore use MILP solver CPLEX to attempt to generate proven optimal solutions for it. However, we were met with mixed results in doing so. Typically, to accomplish that task, CPLEX would require much larger CPU times than we were prepared to pay. Settling for reduced CPU times, quite frequently resulted in nonnegative columns being generated or previously generated columns being generated yet again.

To avoid generating repeated columns, we have devised a disjunctive cut to be appended to $SP(k)$. To describe it, let $\hat{z}^k$ denote, at iteration j, the integral valued part of the solution returned by $SP(k)$. Let $(\hat{z}^k)_v$ identify component v of $\hat{z}^k$ and define sets $G_0 = \{v : (\hat{z}^k)_v = 0\}$ and $G_1 = \{v : (\hat{z}^k)_v = 1\}$. Cuts $\sum_{v \in G_1} (z^k)_v \leq |G_1| - y$ and $\sum_{v \in G_0} (z^k)_v \geq 1 - y$, where $y \in [0, 1]$, therefore eliminate the previously generated solution from the feasibility set associated with $SP(k)$. These cuts also satisfy the relation: $1 - \sum_{v \in G_0} (z^k)_v \leq y \leq |G_1| - \sum_{v \in G_1} (z^k)_v$, thus leading to the valid disjunctive cut

$$-\sum_{v \in G_0} (z^k)_v + \sum_{v \in G_1} (z^k)_v \leq |G_1| - 1.$$

For the time being, we avoided implementing an implicit enumeration scheme for our decomposition algorithm. This is done because, under acceptable CPU times, stopping the column generation procedure at an early stage allow us, as we explain next, to obtain good quality feasible solutions to MELP. This is confirmed after comparing corresponding solution values, i.e., MELP upper bounds, with the valid lower bounds our decomposition algorithm returns for that problem. It is accomplished by solving to proven integral optimality, using a MILP solver, the RMP in hand, with all columns previously generated.

7 COMPUTATIONAL EXPERIMENTS

In order to test our Dantzig-Wolfe decomposition algorithm, we concentrate on the real world MELP instance investigated in [8]. Let us denote it, Case 1. That instance represents the Brazilian power generation system and was compiled to validate the model discussed in [1]. All necessary input data, such as, among others, investment and operations costs, project capacities and critical and average energy levels are detailed in [8].

Case 1 comprises a portfolio of more than a thousand candidate projects, spanning a planning horizon of 25 years. Cases 2 and 3, that are also used in our experiments, are generalizations of Case 1, under some specific conditions. Namely, Case 2 refers to Case 1 under three loads curves for the electric energy demand and a single seasonal period. Analogously, Case 3 refers to Case 2 under two seasonal periods.

Corresponding dimensions for the original (2)-(9) and the extended formulation (31)-(43), for these three cases, are indicated in Table 1, that comprises the number of integer and continuous variables, and the number of constraints.

Model	Case	Int.Var	Cont.Var	Constraints
Original	1	15244	56834	137171
	2	15244	170502	322419
	3	15244	341004	620885
Extended	1	15244	221777	198701
	2	15244	335445	383949
	3	15244	505947	682415

Table 1: Problems Dimensions

Corresponding computational results, using the extended formulation, for different fixed numbers of column generation rounds, are reported in Table 2, where the relative duality gap and total time are given in % and hours, respectively.

# Iter.	Case 1		Case 2		Case 3	
D-W	Gap	Time	Gap	Time	Gap	Time
25	1.64	0.7h	1.58	1.7h	1.80	2.5h
50	1.42	1.5h	1.21	3.5h	1.17	5.0h
75	0.96	2.3h	0.92	5.0h	1.34	8.0h
100	0.86	3.3h	0.75	7.0h	1.02	11.0h

Table 2: Computational Results: Extended Formulation

The computational results we report indicate that relatively small duality gaps are associated with the MELP upper bounds we generate. Furthermore, these gaps are considered very acceptable for the planning purposes they are destined for. In any case, independent of that, an optimality guarantee of 1% or less, as the ones we obtain, appear attractive for such a difficult and large scale problem as the one we address.

Finally, under the original MELP formulation (1)-(9) we previously used, solving to proven integral optimality, using a MILP solver, a relative duality gap of approximately 0.50% was attained after 3 hours for Case 1 and some days for Case 2. Concerning the third instance, even solving the LP relaxation through an IPA, proved unattainable, due to matrix fill-up.

8 CONCLUDING REMARKS

In this paper, we introduce a Dantzig-Wolfe decomposition algorithm for the long term power generation and distribution planning problem. The algorithm was tested on real-world instances representing the Brazilian power generation system over a 25 years planning horizon. The computational results we obtained indicate that our algorithm could be used as an effective planning tool.

For the future, we plan to further enhance that algorithm by implementing an implicit enumeration scheme associated with it. The idea being not to necessarily find proven optimal MELP solutions but to quickly generate very near optimal ones, guided by an enumeration scheme.

REFERENCES

[1] Maria L.V. Lisboa, Luiz Guilherme B. Marzano, Carlos Henrique M. Sabóia, Maria Elvira P. Maceira and Albert C.G. de Melo. ”A mixed-integer programming model for long term generation expansion planning of the Brazilian system”. 16th Power System Computation Conference, 2008, Glasgow.

[2] Maria L.V. Lisboa, Luiz Guilherme B. Marzano, Carlos Henrique M. Sabóia and J.M. Damázio. ”Incorporation of the Natural Gas System into MELP model”, in Portuguese, Proceedings of XI Symposium of Specialists in Electric Operational and Expansion Planning (XI SEPOPE), Belém, Brazil, 2009.

[3] PNE 2030 - Brazilian National Energy Plan 2030 (in Portuguese). Brazilian Ministry of Mines and Energy. November 2007.

[4] Kavinesh J. Singh, Andy B. Philpott and R. Kevin Wood. ”Dantzig-Wolfe Decomposition for Solving Multi-Stage Stochastic Capacity-Planning Problems”. Operations Research, Vol. 57, No. 5, pp. 1271-1286, 2009.

[5] Krarup, J. and O. Bilde. ”Plant location, set covering and economic lot size: An O(*mn*)-algorithm for structured problems”. *Numerische Methoden bei Optimierungsaufgaben, Band 3 (Tagung, Oberwolfach, 1976)*, Vol. 36. Internat. Ser. Numer. Math. Birkhäuser, Basel, Switzerland, 155-180.

[6] Arun P. Sanghvi and Ira H. Shavel. ”Investment Planning for Hydro-Thermal Power System Expansion: Stochastic Programming Employing the Dantzig-Wolfe Decomposition Principle”. IEEE Transactions on Power Systems, Vol. PWRS-1, No. 2, May 1986.

[7] Paul Wentges. ”Weighted Dantzig-Wolfe Decomposition for Linear Mixed-Integer Programming”. International Transactions in Operational Research, Vol. 4, No. 2, pp. 151-162, 1997.

[8] Maria L.V. Lisboa, Luiz Guilherme B. Marzano, Carlos Henrique M. Sabóia, Maria Elvira P. Maceira and Albert C.G. de Melo. ”Validação do modelo MELP para utilização no Plano Nacional de Energia”. Proceedings of XIX SNPTEE, 2007, Rio de Janeiro, Brazil.