# Solving the unit commitment problem of hydropower plants via Lagrangian Relaxation and Sequential Quadratic Programming

ERLON C. FINARDI[1], EDSON L. DA SILVA[1]
and CLAUDIA SAGASTIZÁBAL[2]

[1]LabPlan – Laboratório de Planejamento de Sistemas de Energia Elétrica
Campus Universitario, Trindade, Florianópolis – 88040-900, SC, Brasil
[2]IMPA – Instituto de Matematica Pura e Aplicada
Estrada Dona Castorina 110, Jardim Botânico, Rio de Janeiro – 22460-320, RJ, Brasil
E-mail: erlon@labplan.ufsc.br / edson@labplan.ufsc.br / sagastiz@impa.br

**Abstract.** We consider the optimal scheduling of hydropower plants in a hydrothermal inter-connected system. This problem, of outmost importance for large-scale power systems with a high proportion of hydraulic generation, requires a detailed description of the so-called hydro unit production function. In our model, we relate the amount of generated hydropower to nonlinear tailrace levels; we also take into account hydraulic losses, turbine-generator efficiencies, as well as multiple 0-1 states associated with forbidden operation zones. Forbidden zones are crucial to avoid nasty phenomena such as mechanical vibrations in the turbine, cavitation, and low efficiency levels. The minimization of operating costs subject to such detailed constraints results in a large-scale mixed-integer nonlinear programming problem. By means of Lagrangian Relaxation, the original problem is split into a sequence of smaller and easy-to-solve subproblems, coordinated by a dual master program. In order to deal better with the combinatorial aspect introduced by the forbidden zones, we derive three different decomposition strategies, applicable to various configurations of hydro plants (with few or many units, which can be identical or different). We use a Sequential Quadratic Programming algorithm to solve nonlinear subproblems. We assess our approach on a real-life hydroelectric configuration extracted from the south sub region of the Brazilian hydrothermal power system.

**Mathematical subject classification:** 90C90, 46N10, 47N10.

**Key words:** hydrothermal systems, Unit Commitment Problems, Lagrangian Relaxation, sequential quadratic programming.

## 1 Introduction

The optimal generation scheduling is an important daily activity for electric power generation companies. The goal is to determine which units are to be used in order to generate enough power to satisfy demand requirements and various technological constraints, with minimum operating cost. In particular, hydrothermal systems must consider the stream-flow equations for reservoirs. These equations couple all the reservoir along a hydro-valley, because the amount of outflow water[1] released by one power plant affects water volumes in all the plants downstream. Furthermore, water travel times and alternative uses of water, such as irrigation or flood control, for example, must also be taken into account.

The optimal scheduling of hydropower plants is called the Hydro Unit Commitment (HUC) problem. To solve the HUC problem, a highly sophisticated modeling for the operation of hydro plants is required. Specifically, a hydropower plant may be composed of several turbine-generator groups, referred to in this work as "units". The amount of power generated by one hydro unit depends on the efficiency of both the turbine and the generator, as well as on the net head and the unit turbined outflow. In turn, the net head is a nonlinear function of the storage and of the reservoir outflow. The joint turbine-generator efficiency varies with the net water head and the unit turbined outflow. In addition, the existence of forbidden operation regions prevents the unit from generating power in a wide and continuous range. These regions, modeled by 0-1 variables, aim at avoiding vibrating modes that may produce unwanted power oscillations, cavitation phenomena, and low levels of efficiency. Thermal power plants have simpler production functions, but they need start-up and shut-down times and they often present nonlinear operating costs.

As a result, the hydrothermal unit commitment problem is a large-scale mixed-integer nonlinear programming problem which can only be effectively solved by

---

[1]Turbined and spilt water flow.

applying decomposition techniques. Lagrangian Relaxation (LR) is particularly suitable for this type of problems, [1-5], although some other methodologies have also been proposed [6]. However, so far none of the works in the area has considered a modeling as comprehensive as ours, with representation of hydraulic losses, nonlinear tailrace levels, turbine-generator efficiencies, and forbidden operation zones.

With respect to the solution method, our contribution consists in a thorough analysis of three different decomposition schemes, all derived from LR. The first strategy relies on a complete enumeration of all possible 0-1 operating states of the units composing a hydropower plant. This approach is suitable for plants with a low number of identical units. The second strategy, requiring less computation effort, is applicable for plants with many units that are different and have many forbidden regions. The third strategy combines the two other approaches, and can be used in systems with both types of hydro plants.

Many of the subproblems resulting from our decomposition schemes are Nonlinear Programs (NLP) of small size. We solve them by a Sequential Quadratic Programming (SQP) method [7,8], in a quasi-Newton variant [9,10], which presents good convergence properties.

Our work is organized as follows. In Section 2 we give the hydro plants and units modeling. Section 3 is devoted to the mathematical formulation of the optimal scheduling problem. The solution strategy, with the three decomposition schemes, is given in Section 4. In Section 5 we report numerical results on a hydrothermal system corresponding to Brazil's southern electric sub region. We end in Section 6 with some concluding remarks.

## 2   Hydro generating units

For a unit $j$, the generated power, $ph_j$, expressed in [MW], depends on the unit turbined outflow, $q_j$, and on the net water head, $hl_j$, and on joint turbine and generator efficiency, $\eta_j$:

$$ph_j = 9,81' \ 10^{-3} h_j hl_j q_j \,. \tag{1}$$

The net water head has the expression:

$$hl_j = fcm - fcj(Q,s) - k_j q_j^2 \,. \tag{2}$$

Here, *fcm* stands for the forebay level. For short term horizon problems, as in our case, the forebay level remains practically constant, specially in the Brazilian case, whose huge reservoirs have typical regularization levels of a couple of years. Therefore, we consider *fcm* constant. By contrast, the downstream level $fcj(.)$ varies abruptly in short times, mainly due to the plant turbined outflow, $Q$, given by the addition of the outflows of all the units composing the plant. For some power plant configurations, *fcj* also varies with the reservoir spillage, $s$. In (2), the term $k_j q_j^2$ represents hydraulic losses resulting from friction of the water in penstock, where $k_j$ is a constant expressed in s²/m⁵ [11].

The unit efficiency, depending on $hl_j$ and $q_j$, is usually represented by hill diagrams given by the factory; see Figure 1. We estimate it by interpolation; see [12], using a polynomial function:

$$\eta_j = \rho_{0j} + \rho_{1j}q_j + \rho_{2j}hl_j + \rho_{3j}hl_jq_j + \rho_{4j}q_j^2 + \rho_{5j}hl_j^2, \qquad (3)$$

where the coefficients $\rho_{0j}, \ldots, \rho_{5j}$ have been computed beforehand.

Figure 1 also displays some important operating constraints on the turbine-generator group. For example, for net head values smaller than the so-called nominal level (41,5m), the turbine is unable to make the generator attain its nominal power (120 MW). On the other hand, for values higher than 41,5m, there is a limit of power limit imposed by the generator capabilities, because the turbine could effectively reach power levels beyond 120 MW. Since at some power levels cavitation phenomena and nasty mechanic vibrations may appear, in order to extend the lifetime of the unit and to avoid power oscillations, such power levels are forbidden. For example, Figure 1 shows a forbidden operation region ranging from 70 to 90 MW.

By combining (1)–(3) we obtain our model for the hydro production function:

$$ph_j = 9,81' \, 10^{-3}\left(\rho_{0j}+\rho_{1j}q_j+\rho_{2j}hl_j+\rho_{3j}hl_jq_j+\rho_{4j}q_j^2+\rho_{5j}hl_j^2\right)hl_jq_j. \quad (4)$$

For the Brazilian case, $fcj(Q, s)$ is represented by a fourth degree polynomial. Therefore, from (4) we see that $ph_j$ is a polynomial[2] of degree 12 on the variables $Q$ and $s$, and of order 7 in the variable $q_j$.

---

[2]Since $hl_j = f\left(q_j^2, Q^4, s^4\right)$, it follows that $\eta_j = f\left(q_j^4, Q^8, s^8\right)$ and, thus, by (1), $ph_j = f\left(q_j^7, Q^{12}, s^{12}\right)$.
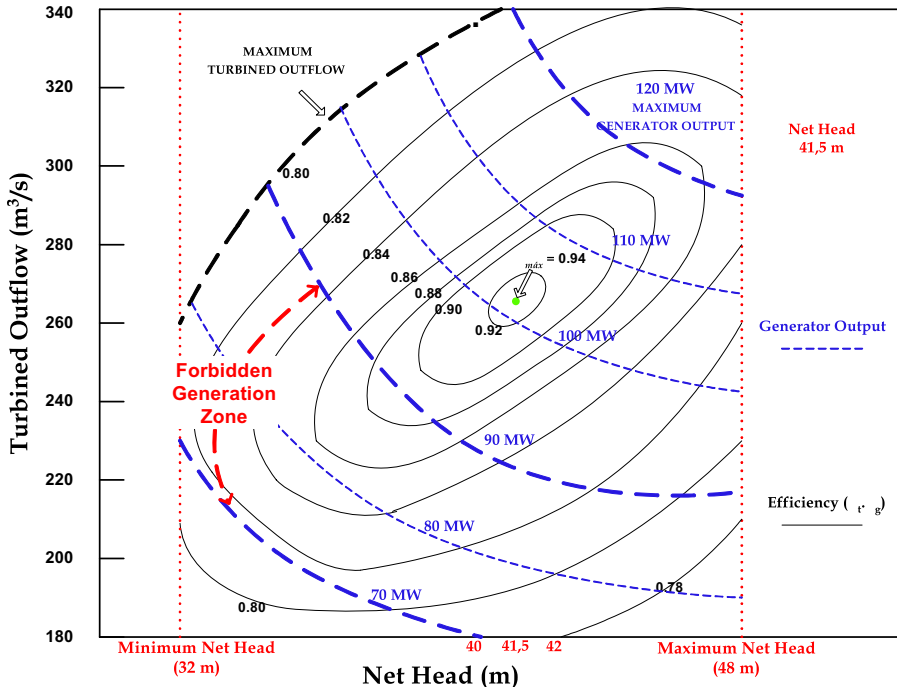
Figure 1 – Hill diagram.

At first sight our model may appear as "too complicated"; however, it is important to realize that only such a detailed description can accurately represent the diverse amounts of power generated by a unit at different operating states.

## 3   Problem formulation

The objective function for the thermal-HUC problem has the expression:

$$\sum_{t=1}^{T} \sum_{i=1}^{I} c_{it}(pt_{it}) + \alpha . \tag{5}$$

Here, the planning horizon is composed by $T$ time steps, the thermal mix has $I$ plants, $c_{it}(.)$ represents the operating cost of the $i$-th thermal plant at time step $t$, and $\alpha$ stands for the system expected future cost at the end of the planning horizon; see (8) below. Frequently $c_{it}(.)$ includes fixed costs as well as fuel costs related to start-up and nominal generation of thermal units [2],[5].

We formulate the thermal-HUC constraints by splitting them into three different subsets, $C_H$, $C_T$ and $C_{HT}$, corresponding to the respective variables involved namely, hydraulic, thermal, or both. Each subset is characterized by a specific type of coupling, such as units in the same power plant along different time steps (time coupling), or different power plants in a given time step (space coupling).

We now proceed to give each constraint in detail.

## 3.1  Constraints involving only hydraulic variables ($C_H$)

• Stream-flow balance equation:

$$v_{r,t+1} + Q_{rt} + s_{rt} - \sum_{m \in \Re_+^{(r)}} \left( Q_{m,t-\tau_{mr}} + s_{m,t-\tau_{mr}} \right) - v_{rt} = y_{rt} . \quad (6)$$

We use the index $r$ for reservoirs, $v$ is the reservoir storage, $y$ is the incremental inflow, $\Re_+^{(r)}$ is a set gathering all reservoirs upstream the $r$-th, and $\tau_{mr}$ is the water travel time between reservoirs $m$ and $r$.

• Maximum $v_r^{max}$ and minimum $v_r^{min}$ storage, and maximum spillage $s_r^{max}$ per reservoir:

$$v_r^{min} \leq v_{rt} \leq v_r^{max} , \qquad 0 \leq s_{rt} \leq s_r^{max} . \quad (7)$$

• Expected future cost function, given by longer term planning models, and estimating the cost of using today water that might become necessary (and expensive) in the future; see [13]. It is a piecewise affine function that depends on the final levels of stocked water, $v_{rT}$:

$$\alpha = f(v_{rT}) . \quad (8)$$

• Penstock water balance equation per reservoir:

$$Q_{rt} = \sum_{j=1}^{J(r)} q_{jrt} , \quad (9)$$

$J(r)$ is the number of generating units in reservoir $r$.

• Power limits, given for each operating region of the unit:

$$\sum_{k=1}^{\Phi_{jr}} ph_{jkrt}^{\min} z_{jkrt} \leq ph_{jrt}(q_{jrt}, Q_{rt}, s_{rt}) \leq \sum_{k=1}^{\Phi_{jr}} ph_{jkrt}^{\max} z_{jkrt}, \qquad (10)$$

$\Phi_{jr}$ denotes the total number of non-forbidden regions of the *j-th* unit in reservoir $r$; $k$ is the corresponding index, and $ph_{jkrt}^{\min,\max}$ stand for the minimum and maximum power limits. The binary variable $z_{jkrt}$ is 1 if the *j-th* unit in reservoir $r$ is operating in the *k-th* region at time step $t$, and it is set to 0 otherwise.

• Reservoir power balance:

$$PH_{rt} = \sum_{j=1}^{J(r)} ph_{jrt}(q_{jrt}, Q_{rt}, s_{rt}). \qquad (11)$$

• Reserve constraints:

$$\left( \sum_{j=1}^{J(r)} \sum_{k=1}^{\Phi_{jr}} ph_{j1rt}^{\max} z_{jkrt} - PH_{rt} \right) \geq rh_{rt}, \qquad (12)$$

$rh_{rt}$ is the minimum reserve of reservoir $r$ at time step $t$.

• Integrality constraints:

$$z_{jkrt} \in \{0, 1\}, \sum_{k=1}^{\Phi_{jr}} z_{jkrt} \leq 1. \qquad (13)$$

In the sequel, to alleviate notation, we write constraints $C_H$ above in the abstract form

$$C_H = C_{HH}(Q, s, V) \quad \cap C_{HUC}(z, q, Q, s, PH),$$

where the vectors $z, q, Q, s, PH$ and $V$ gather the respective variables. The set $C_{HH}$ represents constraints given by (6)-(8), modeling the reservoirs, while $C_{HUC}$ represents the unit constraints, i.e., (9)-(13). In this abstract formulation, $\alpha = \alpha(V)$.

### 3.2 Constraints involving only thermal variables ($C_T$)

- Power limit for each unit:

$$pt_i^{\min} u_{it} \leq pt_{it} \leq pt_i^{\max} u_{it}.$$

  Here $pt_i^{\min,\max}$ stand for the minimum and maximum power limits of unit $i$. The binary variable $u_{it}$ is 1 if the unit is operating at time step $t$, and it is set to 0 otherwise.

- Reserve constraints:

$$pt_i^{\max} u_{it} - pt_{it} \geq rt_{it},$$

  $rt_{rt}$ is the reserve of unit $i$ at time step $t$.

- Minimum up-time, $ti^{up}$, and downtime, $ti^{down}$, for each unit:

$$u_{it} = \left\{ \begin{array}{ll} 1 & \text{if } 1 \leq x_{it} < t_i^{up} \\ 0 & \text{if } -1 \geq x_{it} > -t_i^{down} \\ 0 \text{ or } 1 & \text{otherwise}, \end{array} \right.$$

$$x_{it} = \left\{ \begin{array}{ll} \max(x_{i,t-1}, 0) + 1, & \text{if } u_{it} = 1, \\ \min(x_{i,t-1}, 0) - 1, & \text{if } u_{it} = 0, \end{array} \right.$$

  where the state variable $x_{it}$ is equal to the number of time steps the unit has been up/down until time $t$.

- Ramp constraints:

$$\delta_i(u_{i,t-1}, x_{it}) \leq pt_{it} - pt_{i,t-1} \leq \Delta_i(u_{i,t-1}, x_{it}),$$

  $\delta_i(.)$ and $\Delta_i(.)$ are the maximum allowed variations of generation of the unit between two time steps.

In an abstract formulation, constraints in the set $C_T$ correspond to $C_T(u, pt)$, where $u$ and $pt$ are vectors gathering all binary and continuous thermal variables, respectively.

### 3.3   Constraints involving both hydraulic and thermal variables (C$_{\mathrm{HT}}$)

- Satisfaction of demand, per time step and subsystem:

$$\sum_{i \in I_e} pt_{it} + \sum_{r \in R_e} PH_{rt} + \sum_{l \in \Omega_e} (Int_{let} - Int_{elt}) = D_{et} . \qquad (14)$$

The interconnected hydrothermal system is divided into subsystems, indexed by $e$. Accordingly, all thermal units (reservoirs) of subsystem $e$ are gathered in the index set $I_e(R_e)$. There are $\Omega_e$ subsystems interconnected with subsystem $e$; the exchange of energy at time $t$, is denoted by $Int_{let}$ ($Int_{elt}$ respectively) when it goes from subsystem $l$ to $e$ (from $e$ to $l$, respectively). Finally, $D_{et}$ is the demand of subsystem and at time $t$.

- Subsystems exchange limits, from $e(l)$ to $l(e)$, at time $t$, $Int_{elt}^{\max}$, ($Int_{let}^{\max}$):

$$0 \leq Int_{let} \leq Int_{let}^{\max}, 0 \leq Int_{elt} \leq Int_{elt}^{\max} .$$

In our abstract notation, the set C$_{HT}$ is written as C$_{HT}(pt, PH, Int)$, where the vector $Int$ gathers the subsystem exchanges.

The above description confirms the level of complexity of the optimization problem to be solved. We now address the solution strategy adopted in this work.

## 4   Solving the HUC problem

The economic impact of the optimal scheduling of power plants is undeniable. Because of their solid theoretical background, LR techniques appear in this area as the preferred solution method. In particular, multipliers associated to demand constraints given by (14) are used to price energy.

The *divide to conquer* approach of LR, also called price decomposition [9], is well known. Essentially, coupling constraints are relaxed via Lagrange multipliers whose corresponding dual problem is decomposable into simpler subproblems (called local subproblems). The coordination of subproblems is then done by a master program, which finds new multipliers by making one iteration of a nonsmoth algorithm that maximizes the dual function.

There are many ways of relaxing coupling constraints. An important criterion for deciding how to proceed is the resulting duality gap, which should be the smallest possible. In this matter, the introduction of artificial variables to uncouple constraints appears as a good choice; see [14], and also [5,15] for an application to the thermal UC problem. For this reason, we apply a similar approach in this work, and derive three different decomposition schemes, adapted to different unit configurations in the Brazilian hydrothermal system.

### 4.1    First decomposition strategy – D1

In the abstract notation, the thermal HUC problem becomes:

$$\underset{u, pt, z, q, Q, s, PH, Int}{\text{minimize}} \quad c(pt) + a(V)$$
$$\text{s.t.: } C_T(pt, u) \cap C_{HT}(pt, PH, Int) \cap C_{HH}(Q, s, V) \tag{15}$$
$$\cap C_{HUC}(z, q, Q, s, PH).$$

To achieve decomposition, we introduce artificial variables $pta$ and $PHa$, which duplicate, respectively, $pt$ and $PH$. Variables $pta$ and $PHa$ are used in constraints $C_{HT}$ to replace $pt$ and $PH$. In addition, artificial variables $Qa$ and $sa$ duplicate $Q$ and $s$, respectively. $Qa$ and $sa$ replace $Q$ and $s$ in $C_{HH}$. With these additional variables, (15) is rewritten as follows:

$$\underset{u, pt, pta, z, q, Q, Qa, s, sa, PH, PHa, Int}{\text{minimize}} \quad c(pt) + a(V)$$
$$\text{s.t.: } \quad C_T(pt, u) \cap C_{HT}(pta, PHa, Int) \cap C_{HH}(Qa, sa, V) \tag{16}$$
$$\cap C_{HUC}(z, q, Q, s, PH)$$
$$pt = pta, \quad PH = PHa, \quad Q = Qa, \quad s = sa.$$

In (16) the newly introduced artificial constraints hold the coupling of the problem. Hence, we relax them by associating Lagrange multipliers $\lambda_{PT}$, $\lambda_{PH}$, $\lambda_Q$, $\lambda_S$ and writing the corresponding dual problem[3]:

---

[3]From now on, the Euclidean inner product of two vectors, $\lambda$ and $v$, will be denoted by $\lambda^T v = \Sigma_i \lambda_i v_i$ .

$$\underset{\lambda_{PT},\lambda_{PH},\lambda_Q,\lambda_s}{\text{maximize}} \quad \underset{u,pt,pta,z,q,Q,s,PH,PHa,Int}{\text{minimize}} \quad \left[ c(pt) + \alpha(V) + \lambda_{PT}^{\mathrm{T}}(pt - pta) \right.$$
$$\left. + \lambda_{PH}^{\mathrm{T}}(PH - PHa) + \lambda_Q^{\mathrm{T}}(Q - Qa) + \lambda_S^{\mathrm{T}}(s - sa) \right] \tag{17}$$
$$\text{s.t.:} \quad \mathrm{C_T}(pt, u) \cap \mathrm{C_{HT}}(pta, PHa, Int) \cap \mathrm{C_{HH}}(Qa, sa, V)$$
$$\cap \, \mathrm{C_{HUC}}(z, q, Q, s, PH).$$

Problem (17) can be rewritten as follows:

$$\underset{\lambda=[\lambda_{PT},\lambda_{PH},\lambda_Q,\lambda_S]}{\text{maximize}} \quad \mathrm{D1}(\lambda) := \mathrm{D1_T}(\lambda) + \mathrm{D1_{HT}}(\lambda) + \mathrm{D1_{HH}}(\lambda) + \mathrm{D1_{HUC}}(\lambda), \tag{18}$$

where:

$$\mathrm{D1_T}(\lambda) = \min_{u,pt} c(pt) + \lambda_{PT}^{\mathrm{T}} pt$$
$$\text{s.t.:} \quad \mathrm{C_T}(pt, u), \tag{19}$$

$$\mathrm{D1_{HT}}(\lambda) = \min_{pta,PHa,Int} - \left[ \lambda_{PT}^{T} pta + \lambda_{PH}^{T} PHa \right]$$
$$\text{s.t.:} \quad \mathrm{C_{HT}}(pta, PHa, Int), \tag{20}$$

$$\mathrm{D1_{HH}}(\lambda) = \min_{Qa,sa} \alpha(V) - \lambda_Q^{\mathrm{T}} Qa - \lambda_s^{\mathrm{T}} sa$$
$$\text{s.t.:} \quad \mathrm{C_{HH}}(Qa, sa, V), \tag{21}$$

$$\mathrm{D1_{HUC}}(\lambda) = \min_{z,q,Q,s,PH} \lambda_{PH}^{\mathrm{T}} PH + \lambda_Q^{\mathrm{T}} Q + \lambda_s^{\mathrm{T}} s$$
$$\text{s.t.:} \quad \mathrm{C_{HUC}}(z, q, Q, s, PH). \tag{22}$$

In the LR approach, the primal problem (15) is replaced by the dual problem (18) whose objective function $\mathrm{D1}(\lambda)$, can be split as the sum of four terms, corresponding to subproblems (19)-(22). Subproblem (19) is a nonlinear optimization problem with continuous and binary variables, coupled along time steps, but not along plants. It can be solved by a classic Dynamic Programming method; as in [1],[5]. Subproblem (20) is a standard linear programming (LP) problem, coupled along plants, but not along times steps, which can be solved by any (LP) commercial solver. Subproblem (21) is also an LP problem, coupled both in time and space via the stream-flow constraints given by (6). Even though (21) can be large-scale, an LP solver can still solve it efficiently. Finally, subproblem (22) is a nonlinear mixed-integer optimization problem, uncoupled both in time and

units. This subproblem corresponds to the commitment of hydro units, for a given reservoir and time step. The higher $J(r)$ and $\Phi_{jr}$ (the number of units in the reservoir and of operating zones, respectively) are, the bigger computational effort will be required to solve (22).

Each sub-subproblem in (22), for each time step and for a given power plant, is a mixed-integer NLP problem, with binary variables corresponding to different operating modes in the plant. The total number of possible operating modes is given by the product of all combinations of the operating modes of all the units composing the plant. Each combination of a unit is a configuration where the corresponding binary variables are fixed to one of the feasible values. Once the binary values are fixed, the problem becomes a nonlinear program, whose size is dependent on $J(r)$.

Generally, hydropower plants have identical units, and each unit has a single operating zone. In this case, the total number of modes is no longer $2^{J(r)}$, but $J(r) + 1$ and, thus, a complete enumeration of modes seems a good strategy. Sometimes, however, there are power plants with many different types of units, and several operating modes. For these configurations, an enumeration procedure may become too expensive from the computational point of view. We now introduce an alternative decomposition scheme, adapted to such situations.

### 4.2   *Second decomposition strategy* – D2

In order to avoid the enumerative process required to solve subproblem (22), we eliminate the coupling between the binary variable $z$ and the continuous variables $[q, Q, s]$ which appear in $C_{HUC}$. Therefore, we rewrite (15) as follows:

$$
\begin{aligned}
\underset{u, pt, z, q, Q, s, PH, Int}{\text{minimize}} \quad & c(pt) + \alpha(V) \\
\text{s.t.:} \quad & C_T(pt, u) \cap C_{HT}(pt, PH, Int) \cap C_{HH}(Q, s, V) \\
& \cap C_{HUCa}(q, Q, s, PH) \cap C_{HUCb}(z, q, Q, s) \\
& \cap C_{HUCres}(z, PH).
\end{aligned}
\tag{23}
$$

Now the set $C_{HUCa}$ gathers constraints given by (9) and (11), $C_{HUCb}$ contains (10) and (13) and $C_{HUCres}$ corresponds to the reserve constraint (12). Besides the artificial variables used in (16), we use *pha* to replace the hydro production

function $ph(q, Q, s)$ in the set $C_{HUCb}$, and rewrite (23) as:

$$\underset{u,pt,pta,z,q,Q,Qa,s,sa,PH,PHa,Int}{\text{minimize}} \quad c(pt) + a(V)$$

s.t.:   $C_T(pt, u) \cap C_{HT}(pta, PHa, Int) \cap C_{HH}(Qa, sa, V)$

$$\cap C_{HUCa}(q, Q, s, PH) \cap C_{HUCb}(z, pha)$$

$$\cap C_{HUCres}(z, PH)$$

$$pt = pta, \quad PH = PHa, \quad Q = Qa, \quad s = sa, \quad ph(q, Q, s) = pha.$$

Now not only the artificial constraints, but also the constraint set $C_{HUCres}$ keep the problem coupled. Hence, we relax these constraints by introducing multipliers $\lambda_{PT}, \lambda_{PH}, \lambda_Q, \lambda_S, \lambda_{ph}$ and $\lambda_{Res}$ and writing the dual problem:

$$\underset{\lambda_{PT},\lambda_{PH},\lambda_Q,\lambda_s}{\text{maximize}} \quad \underset{u,pt,pta,z,q,Q,Qa,s,sa,PH,PHa,Int}{\text{minimize}}$$

$$\left[ c(pt) + \alpha(V) + \lambda_{PT}^T(pt - pta) + \lambda_{PH}^T(PH - PHa) \right.$$

$$+ \lambda_Q^T(Q - Qa) + \lambda_S^T(s - sa) + \lambda_{ph}^T[ph(q, Q, s) - pha]$$

$$\left. + \lambda_{Res}^T[PH - ph^{max}z + rh] \right] \tag{24}$$

s.t.:   $C_T(pt, u) \cap C_{HT}(pta, PHa, Int) \cap C_{HH}(Qa, sa, V)$

$$\cap C_{HUCa}(q, Q, s, PH)$$

$$\cap C_{HUCb}(z, pha).$$

Here, $ph^{max}$ and $rh$ are vectors corresponding to a unit's maximum capacity and to reserve levels for the plant, respectively. Separability in (24) has now the expression:

$$\underset{\lambda=[\lambda_{PT},\lambda_{PH},\lambda_Q,\lambda_S,\lambda_{ph},\lambda_{Res}]}{\text{maximize}} \quad D2(\lambda) := D1_T(\lambda) + D1_{HT}(\lambda) + D1_{HH}(\lambda)$$

$$+ D2_{HUCa}(\lambda) + D2_{HUCb}(\lambda) + \lambda_{Res}^T rh$$

where $D1_T(\lambda)$, $D1_{HT}(\lambda)$, $D1_{HH}(\lambda)$ are the dual functions from (19), (20) and (21). The remaining terms are:

$$D2_{HUCa}(\lambda) = \underset{q,Q,s,PH}{\min} \lambda_{Res}^T PH + \lambda_{ph}^T ph(q, Q, s) + \lambda_Q^T Q + \lambda_s^T s$$

$$\text{s.t.:} \quad C_{HUCa}(q, Q, s, PH), \tag{25}$$

$$D2_{HUCb}(\lambda) = \underset{z,pha}{\min} - \left[ \lambda_{ph}^T pha + \lambda_{Res}^T ph^{max}z \right]$$

$$\text{s.t.:} \quad C_{HUCb}(z, pha). \tag{26}$$

Subproblem (25) is an NLP problem (with only continuous variables), for each time step and power plant. Subproblem (26) is a mixed-integer linear program on variables corresponding to a single unit, which can be solved by enumeration of the operating zones.

### 4.3   Third decomposition strategy – D3

Our last decomposition combines D1 and D2. Accordingly, we employ D1 for those plants with a reduced number of units and operating zones, while D2 is applied for plants for which D1 would be too expensive. We split the hydro plants index set into $R(\Gamma)$ and $R(\Xi)$, corresponding, respectively, to plants where (22) and (25)-(26) are applied:

$$
\begin{aligned}
&\underset{u,pt,z,q,Q,s,PH,Int}{\text{minimize}} \quad c(pt) + a(V)\\
&\text{s.t.:} \quad C_T(pt, u) \cap C_{HT}(pt, PH, Int) \cap C_{HH}(Q, s, V)\\
&\qquad\qquad \cap C_{HUC}^{r\in R(\Gamma)}(z, q, Q, s, PH) \cap C_{HUCa}^{r\in R(\Xi)}(q, Q, s, PH)\\
&\qquad\qquad \cap C_{HUCb}^{r\in R(\Xi)}(z, q, Q, s) \cap C_{HUCres}^{r\in R(\Xi)}(z, PH)\,.
\end{aligned}
$$

We proceed like for D1 and D2, but splitting the reservoir index sets:

$$
\begin{aligned}
&\underset{u,pt,pta,z,q,Q,Qa,s,sa,PH,PHa,Int}{\text{minimize}} \quad c(pt) + a(V)\\
&\text{s.t.:} \quad C_T(pt, u) \cap C_{HT}(pta, PHa, Int) \cap C_{HH}(Qa, sa, V)\\
&\qquad\qquad \cap C_{HUC}^{r\in R(\Gamma)}(z, q, Q, s, PH) \cap C_{HUCa}^{r\in R(\Xi)}(q, Q, s, PH) \qquad (27)\\
&\qquad\qquad \cap C_{HUCb}^{r\in R(\Xi)}(z, q, Q, s) \cap C_{HUCres}^{r\in R(\Xi)}(z, PH)\\
&pt = pta, \quad PH = PHA, \quad Q = Qa, \quad s = sa,\\
&ph(q, Q, s) = pha, r \in \Gamma(\Xi)\,.
\end{aligned}
$$

After relaxation, the dual problem of (27) is:

$$
\begin{aligned}
\underset{\lambda=[\lambda_{PT},\lambda_{PH},\lambda_Q,\lambda_S,\lambda_{ph},\lambda_{Res}]}{\text{maximize}} \quad D3(\lambda) \ &:= D1_T(\lambda) + D1_{HT}(\lambda) + D1_{HH}(\lambda)\\
&\quad + D1_{HUC}(\lambda) + D2_{HUCa}(\lambda) \qquad (28)\\
&\quad + D2_{HUCb}(\lambda)
\end{aligned}
$$

The dual functions $D1_T(\lambda)$, $D1_{HT}(\lambda)$, $D1_{HH}(\lambda)$ are those in (19), (20) and (21), respectively. The dual function $D1_{HUC}(\lambda)$ from (22) only applies for reservoirs

with index $r \in R(\Gamma)$. The remaining dual functions in (28) apply to reservoirs $r \in R(\Xi)$, and are given by subproblems (25)-(26).

## 4.4   Nonlinear programming subproblems

A crucial issue for an effective application of LR is the fast resolution of subproblems defining the dual function for a fixed multiplier $\lambda$. Since many of such subproblems are nonlinear programs, we implemented a Sequential Quadratic Programming (SQP) quasi-Newton method. More precisely, each iteration $k$ of the algorithm generates a direction $p^k$ by solving the quadratic programming problem:

$$
\begin{array}{ll}
\underset{p}{\text{minimize}} & \nabla f(x^k)^{\mathrm{T}} p + 0.5 (p)^{\mathrm{T}} M^k p \\
\text{s.t.:} & \nabla c_e(x^k)^{\mathrm{T}} p^k + c_e(x^k) = 0, \qquad \nabla c_i(x^k)^{\mathrm{T}} p^k + c_i(x^k) \le 0.
\end{array}
\tag{29}
$$

Here, $f(x^k)$, $c_e(x^k)$ and $c_i(x^k)$ represent, respectively, the objective and equality and inequality constraint functions at a point $x^k$. The matrix $M^k$ estimates the Lagrangian Hessian for the NLP, $L^k$. In order to avoid the calculation of second-order derivatives, and to preserve positive definiteness of the sequence of quasi-Newton matrices, we use a BFGS (Broyden-Fletcher-Goldfarb-Shanno) [16], formula, appended with a Powell correction [17]:

$$
M^{k+1} = M^k - \frac{M^k s^k (s^k)^{\mathrm{T}} M^k}{(s^k)^{\mathrm{T}} M^k s^k} + \frac{r^k (z^k)^{\mathrm{T}}}{(s^k)^{\mathrm{T}} r^k} ,
$$

where:

$$
s^k = x^{k+1} - x^k, \qquad z^k = \nabla_x L^{k+1} - \nabla_x L^k, \qquad r^k = \theta^k z^k + (1 - \theta^k) M^k s^k,
$$

$$
\theta^k = \begin{cases}
1 & \text{if} \quad (s^k)^{\mathrm{T}} z^k \ge 0, 20 (s^k)^{\mathrm{T}} M^k s^k \\[2ex]
\dfrac{0,8 (s^k)^{\mathrm{T}} M^k s^k}{(s^k)^{\mathrm{T}} M^k s^k - (s^k)^{\mathrm{T}} z^k} & \text{if} \quad (s^k)^{\mathrm{T}} z^k < 0, 20 (s^k)^{\mathrm{T}} M^k s^k.
\end{cases}
$$

Globalization of the method is achieved by performing a line search on the following function[4]:

$$
\phi(x^k, \sigma) = \phi(x^k) + \sigma^k \left\| c(x^k)^{\#} \right\|_{\infty} ,
\tag{30}
$$

---

[4]The symbol $^{\#}$ is used to denote only active constraints at $x^k$.

known as Han's merit function [9]. Since $\phi(x^k, \sigma^k)$ is an exact penalty function, there is a finite positive value $\underline{\sigma}^k$ such that an unconstrained minimum of $\phi(x^k, \sigma^k)$ solves the original NLP for all $\sigma^k \geq \underline{\sigma}^k$. We update the parameter $\sigma^k$ accordingly. More precisely, the directional derivative of the merit function along the direction satisfies the relation:

$$
\begin{aligned}
D[\phi(x^k, \sigma^k); p^k] &\leq \nabla f(x^k)^{\mathrm{T}} p^k - \sigma^k \left\| c(x^k)^{\#} \right\|_{\infty} \\
&\leq (p^k)^{\mathrm{T}} M^k p^k + (\zeta^k)^{\mathrm{T}} c(x^k) - \sigma^k \left\| c(x^k)^{\#} \right\|_{\infty} ,
\end{aligned}
\tag{31}
$$

where $\zeta$ is the Lagrange multiplier associated with constraints in (29). For any stationary point of (30), such as $p^k$, it can be shown that the estimate in (31) gives a descent direction for $\phi$ if $M^k$ is positive definite and $\sigma^k$ is updated in order to satisfy:

$$
\sigma^k \geq \left\| \zeta^k \right\|_{\infty} + \overline{\delta}, \qquad \overline{\delta} > 0.
$$

We exit the line search when the Armijo [18] condition is satisfied:

$$
\phi^k(x^k + \alpha p^k) \leq \phi^k(x^k) + \omega \alpha \Delta^k,
\tag{32}
$$

here, $\omega \in ]0, 1/2[$ and $\alpha$ is the positive stepsize. Ideally, $\Delta^k$ should be the exact value of the directional derivative. We estimate it by the upper bound from (31):

$$
\Delta^k := \nabla f(x^k)^{\mathrm{T}} p^k - \sigma^k \left\| c(x^k)^{\#} \right\|_{\infty} .
$$

Finally, we add an extra term to $p^k$ in order to avoid Maratos effect [19]. This phenomenon may impair the superlinear local convergence rate by rejecting unit stepsizes when close to a solution. The corrected direction $p^k$, as shown in [9], ensures that asymptotically there is enough constraint reduction; see 5.1 below for more details.

## 5   Numerical Results

We assess the three decomposition schemes on a real-life hydroelectric config- uration extracted from the Southern region of the Brazilian hydrothermal power system. More precisely, we consider a system with 121 generating units whose maximum installed capacity is 31.129,2 MW.[5] Figure 2 reports the data for the

---

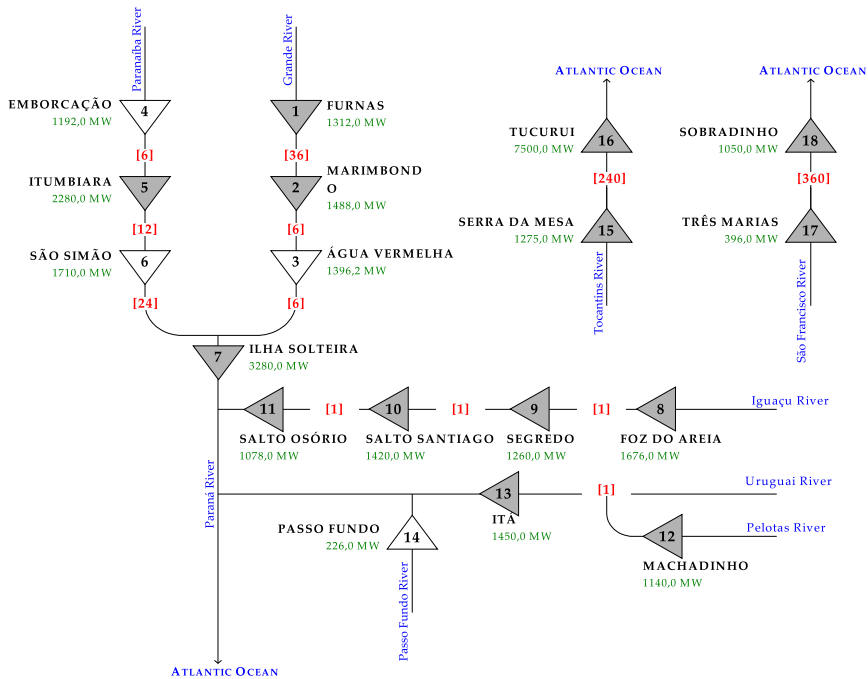[5]This amount corresponds to about 49% of the total hydraulic capacity of Brazil.

Figure 2 – Hydroelectric configuration.

system, where power plants numbered #3, #4, #6 and #14 have production functions independent of spillage. Values between brackets in Fig. 2 correspond to water travel times, expressed in hours. For this configuration, the biggest power plants are #7 and #16, with 20 units each one, while the smallest plant, #14, has only 2 units. The planning horizon of two days is discretized in hourly time steps, yielding $T = 48$. Initial reservoir volumes were taken at 50% of the usable volumes, while the inflows were considered null. We do not address here the dual solution in detail, we refer to [5] for this subject. Instead, we fixed Lagrange multipliers for each reservoir and time step, $\lambda_{PHrt}$, and use a proximal quasi-Newton variant of a bundle method to optimize the remaining multipliers; see Ch. 9 in [9]. The values for $\lambda_{PHrt}$ are chosen based on generation costs associated with typical demand curves, i.e., with higher values for peak times with high demand[6].

---

[6]These costs presented values between 0 and 45 $/MW.

We implemented the three dual subproblems D1$_{\text{UCH}}$, D2$_{\text{UCH}}$, D3$_{\text{UCH}}$, corresponding respectively to (22), (25)-(26) and (28). For the dual solution we use N1CV2 code; see [20]. Subproblem D1$_{\text{UCH}}$ has $(R + Rv) \times T$ variables, where $Rv$ denotes the number of reservoirs with production function depending on spillage. Since $T = 48$, $R = 18$ and $Rv = 14$, subproblem (22) has 1536 variables. In (25)-(26) subproblem D2$_{\text{UCH}}$ has $(2 \times R + Rv + ngmix) \times T = 8208$ variables, where $ngmix = 121$ denotes the total number of units in the mix. The size of subproblem D3$_{\text{UCH}}$ depends on the decomposition scheme. The LP given by D1$_{\text{HH}}$ (21), as well as other LPs are solved using ILOG CPLEX 7.1 solver. For D1$_{\text{UCH}}$ there are, in addition, 6288 NLP problems. Finally, for D2$_{\text{UCH}}$ there are $R \times T = 864$ NLP and $T \times ngmix = 5808$ easy mixed-integer LP problems[7].

For D1$_{\text{UCH}}$ we found an optimal value of \$ –17.339.230,0, after 175 iterations, that took 180 minutes of CPU times in a Pentium III 550 MHz computer with 128 Mb of RAM memory. For D2$_{\text{UCH}}$ the optimal value found was \$ –17.450.168,0, after 325 iterations in 50 minutes. Since the dual value in D2$_{\text{UCH}}$ is smaller than the one from D1$_{\text{UCH}}$, and dual values give lower bounds for the primal optimal value, we can conclude that primal variables associated with D1$_{\text{UCH}}$ are better than those associated with D2$_{\text{UCH}}$.

To further assess the previous remark, we now consider in more detail some selected primal variables, for some specific reservoirs. Figures 3 and 4 show the values for $Q$ and $Qa$ obtained at the last iteration of both D1$_{\text{UCH}}$ and D2$_{\text{UCH}}$ for Sobradinho #18 power plant. We also show the value of $\lambda_{PHrt}$ (price) used to solve the subproblems.

It can be seen in the figures that $Qa$ takes mostly two values: 0 and 4278 m$^3$/s (maximum value). This *bang-bang* behaviour is explained by the linear nature of subproblem (21). The values for $Q$ exhibit a different behaviour, closer to the price profile, i.e., to $\lambda_{PHrt}$. From the comparison of D1$_{\text{UCH}}$ and D2$_{\text{UCH}}$ we see that the relaxed primal constraint[8] is more violated for D2$_{\text{UCH}}$, a problem that contains a higher number of relaxed constraints. However, infeasibility becomes smaller for time steps with bigger $\lambda_{PHrt}$, i.e., for peaks of demand. For these time steps, primal points obtained with D2$_{\text{UCH}}$ are good approximations to those

---

[7]These are indeed easy to solve problems, with only two variables (one integer, one continuous), and two constraints.
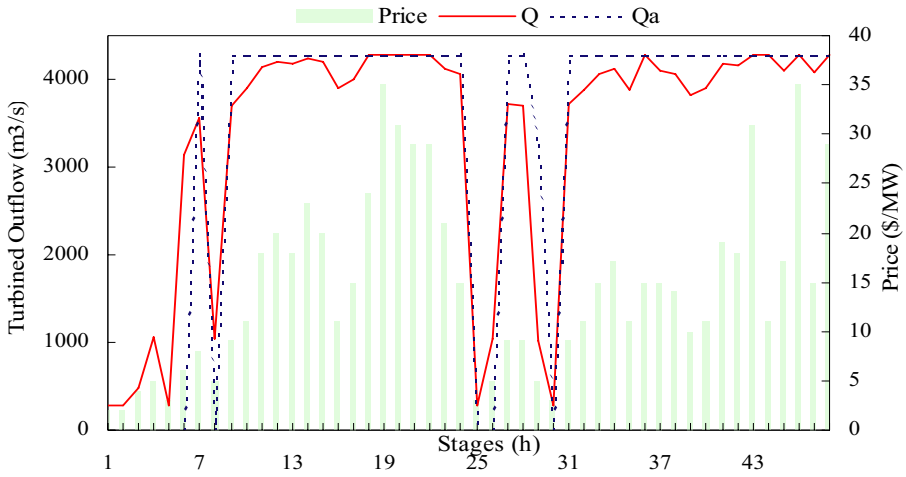
[8]The difference between $Q$ and $Qa$.

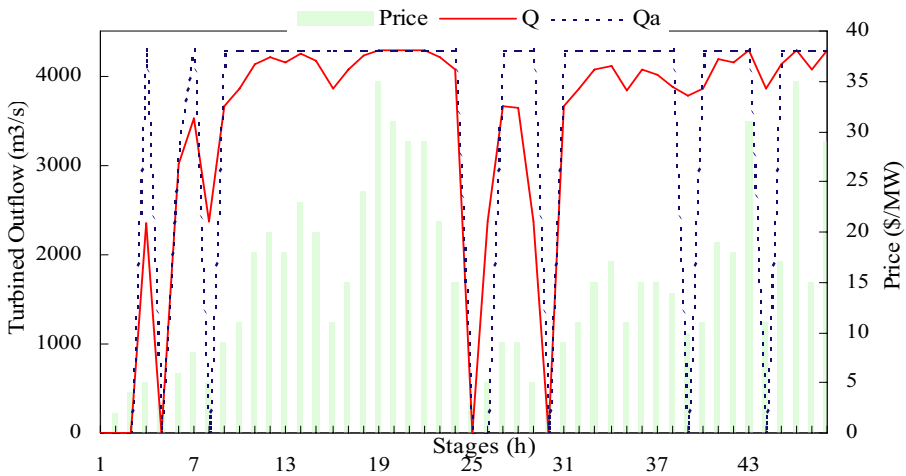Figure 3 – Sobradinho Power Plant #18 $D1_{UCH}$.



Figure 4 – Sobradinho Power Plant #18 $D2_{UCH}$.

from $D1_{UCH}$, requiring a computational effort that can be up to 3 times bigger. For lower prices, both problems give primal points that are even more infeasible, the worse values being associated with $D2_{UCH}$.

The results observed for Sobradinho power plant are typical for all the plants in the mix, with variations in the computed primal infeasibility. In general, we observed that for reservoirs downstream, that tend to operate with outflows near

to the nominal values, primal points obtained from $D2_{UCH}$ were close to those
from $D1_{UCH}$. We report this behaviour in Figures 5 and 6, with the results for
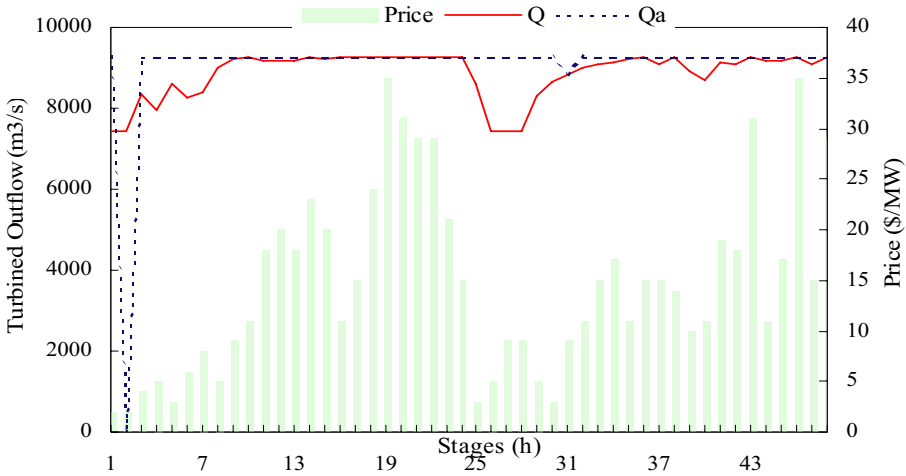Ilha Solteira – 7 power plant.



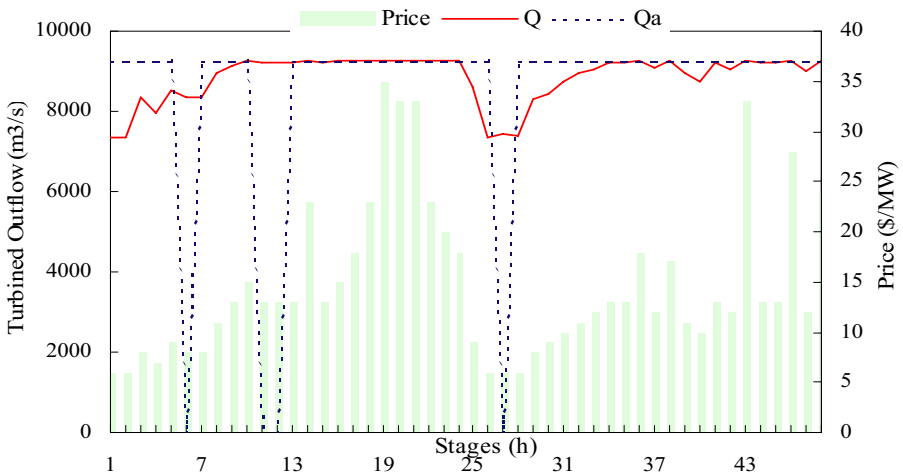Figure 5 – Ilha Solteira Power Plant #7 $D1_{UCH}$.



Figure 6 – Ilha Solteira Power Plant #7 $D2_{UCH}$.

We conclude from our experiments that, in terms of primal solutions, sub-
problem $D1_{UCH}$ is a better option. However, since average hourly CPU times
for $D1_{UCH}$ are 3.5 higher than $D2_{UCH}$, the enumerative process required by

$D1_{UCH}$ should not be used for power plants with complex configurations, such as #7 and #16. In Table 1 we report the main results for the third decomposition scheme, $D3_{UCH}$, where for R $\Xi$ = {7, 16} we applied the scheme corresponding to (25)-(26). We also give, for comparison purposes, the previously obtained values for $D1_{UCH}$ and $D2_{UCH}$.

| Dual Problem | $D1_{UCH}$ | $D2_{UCH}$ | $D3_{UCH}$ |
|:---:|:---:|:---:|:---:|
| Cost ($) | −17.339.230 | −17.450.168 | −17.359.077 |
| Iteration | 175 | 325 | 220 |
| Time (minutes) | 180 | 50 | 70 |
| Variables | 1536 | 8208 | 2592 |
| Number of PNL[9] | 6288 | 864 | 4560 |

Table 1 – Numerical results.

The computational effort required for solving subproblems involving the big plants #7 and #16 (each one having 20 identical units , is clear in Table 1. Even though $D3_{UCH}$[10] needs to solve 27,50% NLP problems less than $D1_{UCH}$, in terms of CPU times the gain was of 61,11%. Another important matter shown by Table 1 concerns dual optimal values: the (absolute value) difference between $D2_{UCH}$ and $D1_{UCH}$ is of $ 110.938, while the difference between $D3_{UCH}$ and $D1_{UCH}$ is $ 19.847, i.e., about 5.5 smaller.

After solving the dual problem, it is necessary to adjust the primal points corresponding to the dual optimum in order to find a primal point that is feasible, i.e., that satisfies the constraints. In order to recover primal feasibility, a purification-like phase should be executed afterwards. Such processes are often based on heuristics depending on the particular problem structure; see for example [21,22]. In particular, for the Brazilian case, general-purpose combinatorial optimization heuristics are not suitable. An augmented Lagrangian technique seems in this case better. We refer to [5] for a description of this technique in a similar context.

---

[10]Ilha Solteira #7 and Tucuruí #16 power plants are the only ones where (22), requiring the enumerative process, was not employed; we use (25)-(26) instead.

## 5.1   Sequential Quadratic Programming Algorithm

In our work, NLP problems have $ng + 2$ variables and $2ng + 2$ constraints, where $ng$ is the number of generating units in the considered configuration (in our example $ng \leq 20$). SQP algorithms do not need starting feasible points; in our implementation starting points only satisfy constraints (6) and (10) (i.e., both the penstock stream-flow equations and power limit constraints), but not the reserve constraint (12). In addition, to prevent the starting quadratic program (29) to be infeasible, we introduced additional constraints on the direction obtained from (29), aimed at satisfying physical operating bounds. Such bounds are related to each unit maximum turbined outflow, as well as maximum spillage levels. We mention that more sophisticated alternatives would be possible, for instance considering active constraints as [23], or introducing slack variables in a trust-region SQP, as in [24].

To solve each quadratic program (29), we use the Fortran code PLCBAS, an active set QP solver described in [25]. Once the direction is computed, the algorithm checks validity of (32) and then performs, if needed, a correction strategy to avoid Maratos effect. A remedy proposed by Maratos in [19] is to move towards the feasible region by using a second order model of the constraints. A more simple strategy is to use the corrected step $p^k$ as follows:

$$p^k = p^k - \underline{A}^k c(x^k + p^k),$$

where $\underline{A}^k$ is a right inverse for the Jacobian matrix whose rows are the transposed gradients of the constraint functions at $x^k$. The correction can be thought of as a constraint restoration step from $x^k + p^k$. It is called 'second-order correction' because $c(x^k + p^k) = O(|p^k|^2)$. Although this modification preserves fast convergence, it may no longer yield a descent direction for the merit function (30). For this reason, instead of searching the stepsize along the line $x^k + \alpha p^k$, we use the arc

$$0 < \alpha| \rightarrow x^k + \alpha p^k - \alpha^2 \underline{A}^k c(x^k + p^k).$$

Since $p^k$ is tangent to the arc at $\alpha = 0$, Armijo condition (32), written with $p^k$ replaced by the corrected step $p^k$, is eventually satisfied whenever $p^k$ is a descent direction, and the "arc" search ends; see Prop. 15.7 in [9]. The additional number of constraint evaluation required by our correction is compensated by

the robustness and efficiency gained by the method. In our runs, at each NLP solution, the second order correction was used in average 30% of the iterations.

We use the following stopping test:

$$opt\_test = \left\| \nabla L^k \right\|_\infty + \left\| c(x^k)^\# \right\|_\infty \leq eps,$$

with $eps = 1, 0 \times 10^{-8}$. Emergency exits, after 300 iterations of 600 simulations were also implemented. The observed average values for achieving convergence were of 8 iterations and 10 simulations.

In order to assess our self-made algorithm, we compare its performances the NLP solver Easy!, available for academic applications from the Optimization Group at IME, University of Campinas, Brazil [26]. This solver uses the augmented Lagrangian method described in [27]. We obtained identical results, with inferior CPU times than those employed by Easy!. However, we did not take advantage of all resources available in Easy! solver[11]. Furthermore, our own implementation was tailored for the structure of our specific problem.

## 6   Concluding Remarks

We address in this work the problem of optimal commitment of hydraulic generating units in a hydrothermal power system. Two main topics were discussed, namely the modeling and solution strategy for the hydraulic problem. We gave a detailed modeling for the production function of each hydro unit, which takes into account the effect of variable efficiency rates, hydraulic losses, tailrace levels, as well as multiple operating zones. With respect to the solution methodology, we gave an LR decomposition approach using variable duplication to uncouple difficult constraints. We assessed the decomposition method by implementing our approach and testing it on a real-life hydraulic configuration, extracted from the Brazilian system. Our implementation focused on the hydraulic subproblems. We analyzed in detail the practical applicability of three decomposition schemes, in terms of CPU times and obtained primal points. For our configuration, the enumerative process appeared preferable for most power plants. For the whole Brazilian system, double in size to our test, this approach might become too

---

[11]Easy! Solver is more efficient when analytic derivatives are available for the simulations; otherwise a finite difference approximation needs to be estimated at each simulation.

expensive and a combined strategy, like the one given in our third decomposition scheme, may be preferable.

## REFERENCES

[1] S. Ruzic, N. Rajakovic and A. Vuckovic, A Flexible Approach to Short-Term Hydro-Thermal Coordination, PART I: Problem Formulation and General Solution Procedure, *IEEE Transactions on Power Systems*, **11**(3), August (1996).

[2] C. Li, E. Hsu, A.J. Svoboda, C.L. Tseng and R.B. Johnson, Hydro Unit Commitment in Hydro-Thermal Optimization, *IEEE Transactions on Power Systems*, **12**(2), May (1997).

[3] X. Guan, A. Svoboda and C.A Li, Scheduling Hydro Power Systems with Restricted Operating Zones and Discharges Ramping Constraints, *IEEE Transactions on Power Systems*, **14**(1), February (1999).

[4] E. Ni, X. Guan and R. Li, Scheduling Hydrothermal Power Systems with Cascaded and Head-Dependent Reservoirs, *IEEE Transactions on Power Systems*, **14**(3), August (1999).

[5] A. Belloni, A. Diniz, M.E.P. Maceira and C. Sagastizábal, Bundle Relaxation and Primal Recovery in Unit Commitment Problems – The Brazilian Case, *Annals of Operations Research Special – Volume on OR models for Energy Policy, Planning and Management*, Kluwer Academic Publishers, Netherlands, 2001.

[6] G.B. Sheble and G.N. Fahd, Unit Commitment Literature Synopsis, *IEEE Transactions on Power Systems*, **9**(1), February (1994).

[7] P.T. Boggs and J.W. Tolle, Sequential Quadratic Programming, *Acta Numerica*, (1996), 1–51.

[8] J. Nocedal and S.J. Wright, *Numerical Optimization*, Springer Series in Operations Research (1999).

[9] J.F. Bonnans, J.C. Gilbert, C. Lemaréchal and C. Sagastizábal, *Numerical Optimization. Theoretical and Practical Aspects*, Universitext, Springer-Verlag, Berlin (2002).

[10] D.P. Bertsekas, *Nonlinear Programming*, Athena Scientific, $2^{nd}$ Edition, Belmont, MA, (1999).

[11] A. Arce, T. Ohishi and S. Soares, Optimal Dispatch of Generating Units of the Itaipú Hydroelectric Plant, *IEEE Trans. Power Systems*, **17**(1), February (2002).

[12] F.J. Heredia and N. Nabona, Optimum Short-Term Hydrothermal Scheduling with Spinning Reserve Through Network Flows, *IEEE Trans. Power Systems*, **10**(3), August (1995).

[13] E.L. da Silva and E.C. Finardi, Parallel Processing Applied To the Planning of Hydrothermal Systems. *IEEE Transactions on Parallel and Distributed Systems*, **14** (2003), 721–729.

[14] C. Lemaréchal and A. Renaud, A Geometric Study of Duality Gaps, With Applications, *Mathematical Programming*, **90**(3) (2001), 399–427.

[15] C. Lemaréchal, F. Pellegrino, A. Renaud and C. Sagastizábal, Bundle Methods Applied to the Unit Commitment Problem, *System Modeling and Optimization*, (1996), 395–402.

[16] S.P. Han, Superlinearly Convergent Variable Metric Algorithms for General Nonlinear Programming Problems, *Mathematical Programming*, **11** (1976), 263–282.

[17] M.J.D. Powell, A Fast Algorithm for Nonlinearly Constrained Optimization Calculations, *Numerical Analysis*, 144–157, Springer-Verlag, Berlin (1978).

[18] L. Armijo, Minimization of Functions Having Lipschitz Continuous First Partial Derivates, *Pacific Journal of Mathematics*, **16**(1-3) (1966).

[19] N. Maratos, Exact Penalty Function Algorithms for Finite Dimensional and Control Optimization, Ph.D. Thesis, University of London (1978).

[20] C. Lemaréchal and C. Sagastizábal, Use of the Code N1CV2, Inria, France (2002).

[21] H. Blondel and A. Renaud, Décomposition Par Les Prix et Lagrangien Augmenté: Une Association Efficace Pour le Traitement des Problèmes de Production, Bulletin EDF, Série C, **4** (1991), 23–43.

[22] F. Zhuang and F.D. Galiana, Towards a More Rigorous and Practical Unit Commitment by Lagrangian Relaxation, *IEEE Transactions on Power Systems*, **3**(2), May (1988).

[23] C.T. Lawrence, A Computationally Efficient Feasible Sequential Quadratic Programming Algorithm, Ph.D. Thesis, University of Maryland, USA (1998).

[24] M.R. Celis, J.E. Dennis and R.A. Tapia, A Trust-Region Strategy for Nonlinear Equality Constrained Optimization, *Numerical Optimization*, SIAM, Philadelphia, 71–82 (1985).

[25] E. Casas and C. Pola, An algorithm for indefinite quadratic programming based on a partial Cholesky factorization, *RAIRO-Recherche Opérationnelle/Operations Research*, **27** (1993), 401–426.

[26] Internet address: http://www.ime.unicamp.br/~martinez/software.htm.

[27] A. Friedlander, J.M. Martinez and S.A. Santos, A New Trust Region Algorithm for Bound Constrained Minimization, *Applied Mathematics and Optimization* **30** (1994), 235–266.