

# Problem of Hydroelectric Resources Management: Numerical Approach

V. A. Bushenkov<sup>1</sup>, M. M. A. Ferreira<sup>2</sup>, A. F. Ribeiro<sup>2</sup> and G. V. Smirnov<sup>3</sup>

<sup>1</sup> CIMA-UE, Department of Mathematics, University of Evora, Portugal

<sup>2</sup> ISR, FEUP, University of Porto, Portugal

<sup>3</sup> Centre of Physics, Department of Mathematics and Applications, University of Minho, Portugal

.....  
...

**Abstract:** In this work we analyze an optimal control problem for a system of two hydroelectric power stations in cascade with reversible turbines. The objective is to optimize the profit of power production while respecting the system's restrictions. Some of these restrictions translate into state constraints and the cost function is nonconvex. This increases the complexity of the optimal control problem. The problem is solved numerically and two different approaches are adopted. These approaches focus on global optimization techniques (Chen-Burer algorithm) and on a projection estimation refinement method (PER method). PER method is used as a technique to reduce the dimension of the problem. Results and execution time of the two procedures are compared.

**Keywords:** Optimal control, global optimization, projection estimation refinement method

## 1 Introduction

Water has becoming a scarce and valuable resource and the concern for an efficient use of it is nowadays more evident. Examples of that are hydro-electric systems equipped with reversible turbines where the same water can be used for different purposes including energy production in different times. Optimal management of multireservoir systems along a river has attracted the interest of many researchers in different contexts (see [9] and references therein). In this work we consider a simplified model for a cascade of two hydro-electric power stations where one of the stations has reversible turbines. Each station is linked to a reservoir and turbines water downstream leading to electric energy production. The presence of reversible turbines enables water to be pumped from a downstream reservoir to be used later on a more convenient time. With this model we address the problem of optimal management of the system, i.e., find the water flows to turbine/pump and the corresponding volumes in the reservoirs that maximize the selling profit of the energy produced in the system.

The problem is formulated as an optimal control problem. The fluxes of water to turbine or pump on each power station are defined by the control variables and the water volumes in reservoirs are the state variables. The

profit of energy sale is the objective function. It is a challenging problem since besides constraints in the control it also involves pure state constraints. Furthermore, the cost function is nonconvex.

The objective in this work is to get a numerical global solution to the problem. After performing a discretization of the optimal control problem, we obtain a maximization problem with an indefinite quadratic cost function subject to linear constraints. The cost function is nonconvex and existence of several local minimum may occur. Thus, application of global optimization methods is convenient. We use the algorithm introduced by Jieqiu Chen and Samuel Burer for minimizing general quadratic forms subject to linear constraints in  $R^n$  (for more details, see [4, 10, 11, 12]).

Two different approaches are discussed and compared. In the first approach, we directly apply the Chen-Burer algorithm. In the second, taking advantage of the specific structure of the cost function, we construct a projection of the set of feasible solutions onto a subspace of the cost function arguments. Such construction makes use of an algorithm based on the Projection Estimation Refinement method, known as PER method (see [1] for details). The dimension of the problem is notably reduced. The Chen-Burer algorithm is then applied to the projected low-dimension problem. The numerical

\* Corresponding author e-mail: [ana.fca.ribeiro@gmail.com](mailto:ana.fca.ribeiro@gmail.com)

solution obtained is used to derive an approximate solution of the original full dimension discrete problem. To that end, we solve a simple convex programming problem. Finally, with the approximate solution as an initial guess, it is used a local optimization method to achieve a numerical solution for the original problem.

A main contribution of this paper resides on the procedure undertaken to improve computation time. Such procedure takes advantage of a specific form of the cost function and involves the combination of several known techniques and algorithms, as described above.

## 2 Problem statement

The problem under consideration is associated to a system with two hydro-electric power stations in a cascade structure. Figure 1 is a representative scheme for such cascade. Each power station is coupled with a reservoir, with reservoirs 1 and 2 identified in the figure by  $V_1$  and  $V_2$  respectively. The arrows represent the capability of power station 1 to pump water from a downstream reservoir. Naturally, this will be done when the price and demand are low.

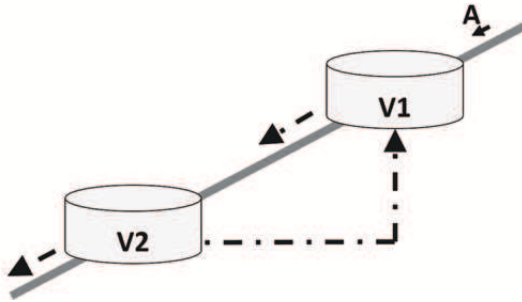


Fig. 1: Cascade with hydro-power stations

For this cascade we assume that the dynamics of water volumes  $V_k(t)$ ,  $k = 1, 2$ , are described by the following control system,

$$\begin{aligned}\dot{V}_1(t) &= A - u_1(t) \\ \dot{V}_2(t) &= u_1(t) - u_2(t),\end{aligned}\quad (1)$$

where the controls  $u_1(t)$  and  $u_2(t)$  represent water flows at time  $t$  that are turbined ( $u_i(t) \geq 0$ ) or pumped ( $u_i(t) < 0$ ) in reservoirs 1 and 2 respectively. The parameter  $A$  denotes the incoming flow which, for simplicity, we take constant. Variable  $u_2(\cdot)$  takes only nonnegative values. Equations (1) are called water balance equations and are present in many references (see, e.g. [13]).

The control and state variables satisfy the following technical constraints:

$$\begin{aligned}V_k(0) &= V_k(T), & \text{for } k = 1, 2 \\ V_k(t) &\in [V_k^m, V_k^M], & \text{for } k = 1, 2 \\ u_k(t) &\in [u_k^m, u_k^M], & \text{for } k = 1, 2.\end{aligned}\quad (2)$$

Here  $V_k^m$  and  $V_k^M$ ,  $k = 1, 2$ , stand for the imposed minimum and maximum water volumes, respectively;  $u_k^m$  and  $u_k^M$ ,  $k = 1, 2$ , are the imposed minimum and maximum for water flows.

The objective is to find optimal controls  $\hat{u}_k(\cdot)$ ,  $k = 1, 2$ , and volumes  $\hat{V}_k(\cdot)$ ,  $k = 1, 2$ , that maximize the cost function,

$$\begin{aligned}J(u(\cdot), V(\cdot)) &= \int_0^T c(t) \left[ u_1(t) \left( \frac{V_1(t)}{S_1} + H_1 - \frac{V_2(t)}{S_2} - H_2 \right) \right. \\ &\quad \left. + u_2(t) \left( \frac{V_2(t)}{S_2} + H_2 \right) \right] dt \rightarrow \max, \quad (3)\end{aligned}$$

where  $c(\cdot)$  is the price of the energy,  $H_k$ ,  $k = 1, 2$ , are the liquid surface elevations and  $S_k$ ,  $k = 1, 2$ , are the areas of the reservoirs. In the integral, the factor multiplied by  $c(t)$  represents potential energy. For simplicity, it is assumed that the reservoirs have cylindric form and that the gravity constant is equal to one. It is also assumed that all the potential energy is converted into electric energy.

We assume that  $c(\cdot)$  takes constant values  $c_1$  and  $c_2$ , respectively on the intervals  $[0, \frac{T}{2}]$  and  $[\frac{T}{2}, T]$ .

Since  $u_1(t) = A - \dot{V}_1(t)$  and  $u_2(t) = u_1(t) - \dot{V}_2(t)$ , the cost  $J(u(\cdot), V(\cdot))$  can be written as

$$\begin{aligned}\int_0^T c(t) \left[ (A - \dot{V}_1(t)) \left( \frac{V_1(t)}{S_1} + H_1 \right) \right. \\ \left. - \dot{V}_2(t) \left( \frac{V_2(t)}{S_2} + H_2 \right) \right] dt.\end{aligned}\quad (4)$$

Without changing the notation, we replace  $J(u(\cdot), V(\cdot))$  by its symmetric. The maximization problem is then transformed into a minimization one. After some calculus, using (2) and neglecting a constant term, we get

$$\begin{aligned}J(u(\cdot), V(\cdot)) &= -\frac{Ac_1}{S_1} \int_0^{T/2} V_1(t) dt - \frac{Ac_2}{S_1} \int_{T/2}^T V_1(t) dt \\ &\quad + H_1(c_2 - c_1)V_1(0) + \frac{c_2 - c_1}{2S_1}V_1^2(0) \\ &\quad - H_1(c_2 - c_1)V_1\left(\frac{T}{2}\right) - \frac{c_2 - c_1}{2S_1}V_1^2\left(\frac{T}{2}\right) \\ &\quad + H_2(c_2 - c_1)V_2(0) + \frac{c_2 - c_1}{2S_2}V_2^2(0) \\ &\quad - H_2(c_2 - c_1)V_2\left(\frac{T}{2}\right) - \frac{c_2 - c_1}{2S_2}V_2^2\left(\frac{T}{2}\right).\end{aligned}\quad (5)$$

### Discretized problem

A discretization of the problem is generated in the following way. Let  $N$  be an even natural number. Define variables  $x$  and  $y$  as

$$x = \left[ V_1(0), V_1\left(\frac{N}{2}\right), V_2(0), V_2\left(\frac{N}{2}\right) \right], \text{ and}$$

$$y = \left[ V_1(1), \dots, V_1\left(\frac{N}{2}-1\right), V_1\left(\frac{N}{2}+1\right), \dots, V_1(N-1), \right. \\ \left. V_2(1), \dots, V_2\left(\frac{N}{2}-1\right), V_2\left(\frac{N}{2}+1\right), \dots, V_2(N-1) \right].$$

The cost function takes the form

$$I(x, y) = \langle a, x \rangle + \langle b, y \rangle + \langle x, Qx \rangle \rightarrow \min, \quad (6)$$

where  $a$  and  $b$  are vectors gathering the linear part of the cost relative to  $x$  and  $y$  respectively and  $Q$  is a matrix defining the quadratic part of the cost function. More precisely,

$$Q = \begin{pmatrix} \frac{c_2 - c_1}{2s_1} & 0 & 0 & 0 \\ 0 & -\frac{c_2 - c_1}{2s_1} & 0 & 0 \\ 0 & 0 & \frac{c_2 - c_1}{2s_2} & 0 \\ 0 & 0 & 0 & -\frac{c_2 - c_1}{2s_2} \end{pmatrix} \quad (7)$$

$$a = \left[ H_1(c_2 - c_1) - \frac{Ac_1}{s_1}, -H_1(c_2 - c_1) - \frac{Ac_2}{s_1}, \right. \\ \left. H_2(c_2 - c_1), -H_2(c_2 - c_1) \right] \quad (8)$$

$$b = \begin{bmatrix} \overbrace{-\frac{Ac_1}{s_1}, \dots, -\frac{Ac_1}{s_1}}^{\frac{N}{2}-1 \text{ components}}, \overbrace{-\frac{Ac_2}{s_1}, \dots, -\frac{Ac_2}{s_1}}^{\frac{N}{2}-1 \text{ components}}, \\ \overbrace{0, \dots, 0}^{\frac{N}{2}-1 \text{ components}}, \overbrace{0, \dots, 0}^{\frac{N}{2}-1 \text{ components}} \end{bmatrix}. \quad (9)$$

The constraints of the problem are translated into conditions below. Conditions (11), (12), (13) and (14) account for control constraints. In the discretization process we use (1) and replace  $u_1(t)$ ,  $u_2(t)$  by  $A - \dot{V}_1(t)$  and  $A - \dot{V}_1(t) - \dot{V}_2(t)$ , respectively. The derivative  $\dot{V}_i(t)$  is replaced by  $V_i(k+1) - V_i(k)$ ,  $i = 1, 2$ .

The amounts  $V_i(T) = V_i(N)$ ,  $i = 1, 2$ , are not included as variables, since  $V_i(N) = V_i(0)$ ,  $i = 1, 2$ . However, we need to ensure the admissibility of control action to go from  $V_i(N-1)$  to  $V_i(N) = V_i(0)$ . This is guaranteed by equations (11) and (12).

For  $k = 0, \dots, N-1$  and  $i = 1, 2$ ,

$$V_i(k) \in [V_i^m, V_i^M], \quad (10)$$

Also,

$$V_1(N-1) + A - V_1(0) \in [u_1^m, u_1^M], \quad (11)$$

$$V_2(N-1) + V_1(N-1) + A - V_1(0) - V_2(0) \in [u_2^m, u_2^M], \quad (12)$$

and for  $k = 0, \dots, N-2$

$$V_1(k) + A - V_1(k+1) \in [u_1^m, u_1^M], \quad (13)$$

$$V_2(k) + V_1(k) + A - V_1(k+1) - V_2(k+1) \in [u_2^m, u_2^M]. \quad (14)$$

The discretized problem is then defined by (6), (10) - (14).

### 3 Numerical methods

The aim of this work is to get a numerical global solution to the minimization of (5) subject to (1) and (2). To this end, we compare in this section two different approaches to solve the associated discretized problem (6), (10) - (14). The following data are considered:

$$\begin{aligned} V_1^m &= 86.7, & u_1^m &= -0.3456, & c_1 &= 2, \\ V_1^M &= 147, & u_1^M &= 0.4392, & c_2 &= 20, \\ V_2^m &= 48.3, & u_2^m &= 0, & H_1 &= 3, \\ V_2^M &= 66, & u_2^M &= 0.8316, & H_2 &= 1, \\ s_1 &= 81.7, & s_2 &= 44.5, & A &= 0.1589. \end{aligned}$$

We take  $N = 24$ .

#### 3.1 1<sup>st</sup> Approach - Chen-Burer Algorithm applied directly

The Chen-Burer algorithm (see [4]) is directly applied to the discretized problem defined by (6), (10) - (14). This algorithm is oriented to global optimization of nonconvex quadratic programming (QP) problems (optimization of a general quadratic function over linear and bounded constraints). It combines a finite branch-and-bound (B&B) scheme, in which branching is based on the

first-order Karush-Kuhn-Tucker (KKT) conditions, with a polyhedral-semidefinite relaxation that is applied at each node of the B&B tree. Such relaxation is derived from completely positive and doubly nonnegative programs. One of the advantages of this method is that the B&B tree is finite. Other, is that we can develop stronger relaxations for the problem (for more details see [4, 10, 11, 12]).

The global optimization solver uses the same syntax as the local optimization routine *QuadProg* of the Matlab and requires an external linear programming solver. The problem must be formulated as:

$$\begin{aligned} \text{minimize} \quad & \frac{1}{2}x^T Hx + f^T x \\ \text{s.t.} \quad & Ax \leq b \\ & A_{eq}x = b_{eq} \\ & LB \leq x \leq UB \end{aligned}$$

where  $x \in \mathbb{R}^n$  is the variable and  $H \in \mathbb{R}^{n \times n}$ ,  $f \in \mathbb{R}^n$ ,  $A \in \mathbb{R}^{m \times n}$ ,  $b \in \mathbb{R}^m$ ,  $A_{eq} \in \mathbb{R}^{m_{eq} \times n}$ ,  $b_{eq} \in \mathbb{R}^{m_{eq}}$ ,  $LB \in \mathbb{R}^n$  and  $UB \in \mathbb{R}^n$  are parameters.

The Chen-Burer algorithm is available at <http://dollar.biz.uiowa.edu/~sburer> under the keyword *QuadprogBB*.

### 3.1.1 Results - Chen-Burer algorithm

Since we take  $N = 24$ , the problem involves 48 variables, 48 boxing constraints and 96 inequality constraints.

The resulting global numerical solution,  $(\hat{x}, \hat{y})$ , is such that

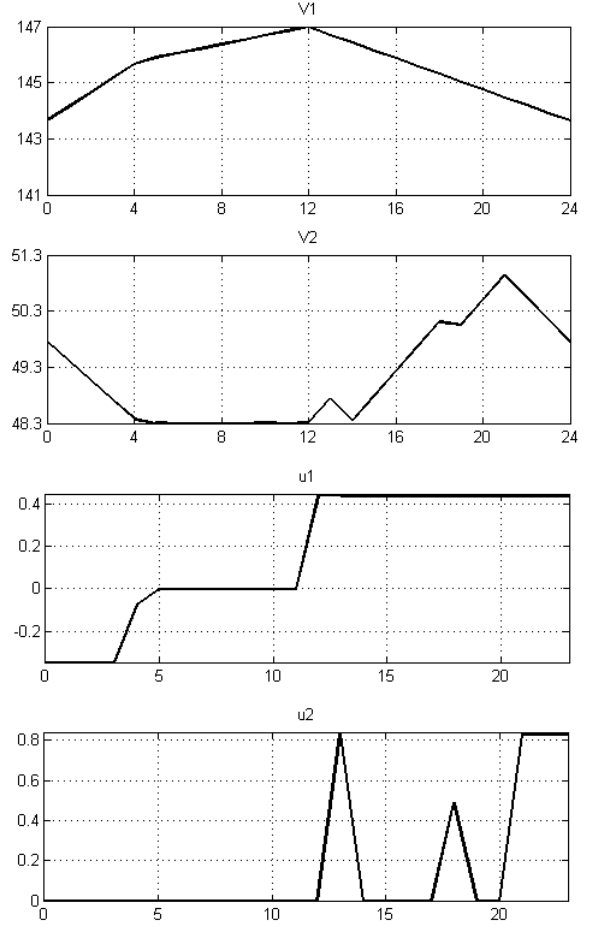
$$\begin{aligned} \hat{x} &= \left[ \hat{V}_1(0), \hat{V}_1\left(\frac{N}{2}\right), \hat{V}_2(0), \hat{V}_2\left(\frac{N}{2}\right) \right] \\ &= [143.64, 147, 49.76, 48.3]. \end{aligned}$$

The associated cost is 308.918€ and the execution time is 24 hours. This computational time revealed too long.

From the numerical solution and using relationships between  $(x, y)$  and  $V_i(k)$ ,  $i = 1, 2$ ,  $k = 0, \dots, 24$ , we construct continuous versions for volume graphics and also for flow graphics. These graphics are shown in figure 2 below.

Observe that in the second half of time interval the behavior of  $V_2(t)$  and  $u_2(t)$  is quite irregular. This can be explained by the existence of many solutions on  $[T/2, T]$ . We will come back to this subject in the end of the section.

Next, it is presented another approach, not so time consuming, to get the global solution for the discretized problem.



**Fig. 2:** Volumes and controls associated to the global solution

### 3.2 2<sup>nd</sup> Approach - Projection Estimation Refinement method

In this section, other approach to solve the discretized problem defined by (6), (10)-(14) is considered. This is motivated by the long computational time taken in the first approach.

We use a specific structure of the cost function that allows us to reduce the dimension of the problem. The set of feasible solutions is projected onto a subspace of the cost function arguments. The Chen-Burer algorithm is applied to the *projected* low-dimensional problem. A solution to the high dimension problem is after obtained by solving a simple convex programming problem.

The projected set is a convex limited set and therefore can be approximated by polytopes with any desired accuracy.

The Projection Estimation Refinement method (PER), described in ([1]), approximates the orthogonal projection  $P$  of a polytope  $X$  onto a subspace by a sequence of

polytopes  $P^0, P^1, \dots, P^k, \dots$  that tend to  $P$ , and  $P^k \subset P$  for all  $k$ . The number of vertices of polytopes increases by one at each iteration. On each iteration a new polytope is constructed on the basis of the previous one using procedures of computing support functions and Fourier-Motzkin convolution method ([2]). In [3] a robust algorithm for solving this problem is proposed.

For approximating polytopes, two descriptions are constructed simultaneously, one as a set of their vertices and the other as a system of linear inequalities,

$$P^k = \{x \in \mathbb{R}^q : \langle c_j, x \rangle \leq d_j, j = 1, 2, \dots, N\}$$

where  $c_j \in \mathbb{R}^q$  and  $d_j \in \mathbb{R}^1$ .

Figures 3 and 4 illustrate two iterations of the PER method applied to a generic convex set  $P$ . The ■-dots are the candidates to include on next step. These are the most distant points of  $P$ , from the faces of  $P_k$ . They are found by maximizing in directions orthogonal to the faces of  $P_k$ . In figure 4 the most distant new point ★-dot is included into  $P_k$ .

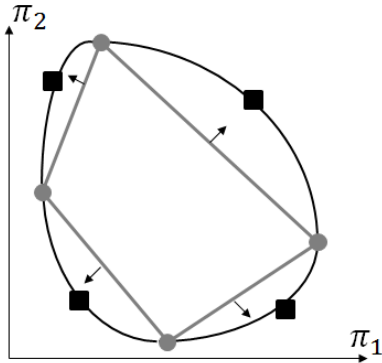


Fig. 3: 1st iteration - the initial set

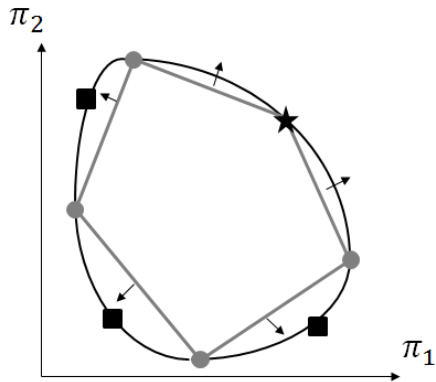


Fig. 4: 2nd iteration - the most distant new point ★ is included into the convex hull

Knowing inequalities of internal approximating sets and the values of the corresponding support functions, it is easy to find external approximating sets  $\bar{P}^0, \bar{P}^1, \dots, \bar{P}^k$ , which contains the set  $P$ , i.e.,  $P^k \subset P \subset \bar{P}^k$  for all  $k$ . This is illustrated in figure 5 below.

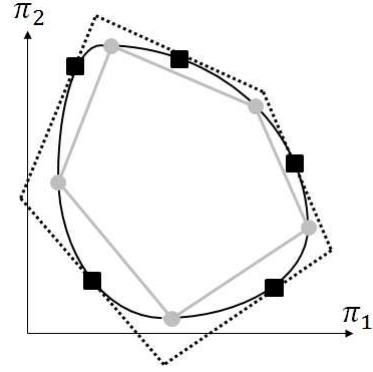


Fig. 5: Internal estimation (convex hull of vertices) and external estimation (described by support-planes)

Computational details and a discussion of these techniques for polyhedral approximations can be found in [6, 8].

Returning to our discretized problem, recall the cost function

$$I(x, y) = \langle a, x \rangle + \langle b, y \rangle + \langle x, Qx \rangle \rightarrow \min,$$

with  $x = [V_1(0), V_1(\frac{N}{2}), V_2(0), V_2(\frac{N}{2})]$ , and

$$y = \left[ V_1(1), \dots, V_1\left(\frac{N}{2}-1\right), V_1\left(\frac{N}{2}+1\right), \dots, V_1(N-1), \right. \\ \left. V_2(1), \dots, V_2\left(\frac{N}{2}-1\right), V_2\left(\frac{N}{2}+1\right), \dots, V_2(N-1) \right].$$

Define a new variable

$$z = \langle b, y \rangle.$$

From this equality, taking into account the definition of  $b$  given in (9), it comes

$$V_1(1) = - \left[ \frac{s_1}{Ac_1} z + V_1(2) + \dots + V_1\left(\frac{N}{2}-1\right) + \right. \\ \left. \frac{c_2}{c_1} \left( V_1\left(\frac{N}{2}+1\right) + \dots + V_1(N-1) \right) \right].$$

The cost function may also be expressed as

$$\langle \bar{a}, \bar{x} \rangle + \langle \bar{x}, \bar{Q}\bar{x} \rangle \rightarrow \min, \quad (15)$$

where  $\bar{x} = (x, z)$ ,  $\bar{a} = (a, 1)$ , and  $\bar{Q} = \begin{pmatrix} Q & 0 \\ 0 & 0 \end{pmatrix}$ , with  $Q$  and  $a$  defined as in (7) and (8).

The projection of the set of feasible solutions onto the subspace of variables

$$(V_1(0), V_1(N/2), V_2(0), V_2(N/2), z)$$

is constructed using PER method algorithm. With the projected set and cost function (15) we obtain an optimization problem in  $R^5$ .

Applying the Chen-Burer algorithm to this lower dimension problem, we get a numerical solution  $(\hat{x}, \hat{z})$ . With  $\hat{x}$  and a simple convex programming problem we deduce an approximate solution  $(\hat{x}, \hat{y})$  to the original discrete problem. Finally, this approximate solution is used as an initial guess in a local optimization software, from which we get a global solution for the original problem.

### 3.2.1 Results with PER method

Using PER method algorithm we get a feasible set for the projected problem (exterior approximation with 18 inequalities and 5 boxing constraints). The Chen-Burer algorithm is applied to this lower dimension problem and the resulting numerical solution is given by

$$\hat{x} = (\hat{x}, \hat{z}) = [140.66, 147, 48.30, 49.16, -68.18].$$

The following convex quadratic programming problem, that we solve using the function *QuadProg* from the Matlab, allows us to get an approximate solution to the original high dimension discretized problem.

$$\begin{aligned} &\text{minimize} && \|\Pi(w) - \hat{x}\|^2 \\ &\text{s.t.} && Aw \leq b \\ &&& LB \leq w \leq UB, \end{aligned}$$

where

$$\begin{aligned} w &= (V_1(0), V_1(1), \dots, V_1(N-1), V_2(0), \\ &\quad V_2(1), \dots, V_2(N-1)), \\ \Pi(w) &= (V_1(0), V_1(N/2), V_2(0), V_2(N/2)), \\ \hat{x} &= (\hat{V}_1(0), \hat{V}_1(N/2), \hat{V}_2(0), \hat{V}_2(N/2)). \end{aligned}$$

and the constraints (10)-(14) are also here considered.

From the resulting numerical solution of this problem we define continuous functions  $V_i(\cdot), u_i(\cdot)$ ,  $i = 1, 2$ . The graphics of these functions are shown in figure 6.

Finally, this global numerical solution can still be used to define controls that jointly with  $\hat{x}$  constitute an initial guess for application of a local optimal control package from [7].

The final result is presented in figure 7.

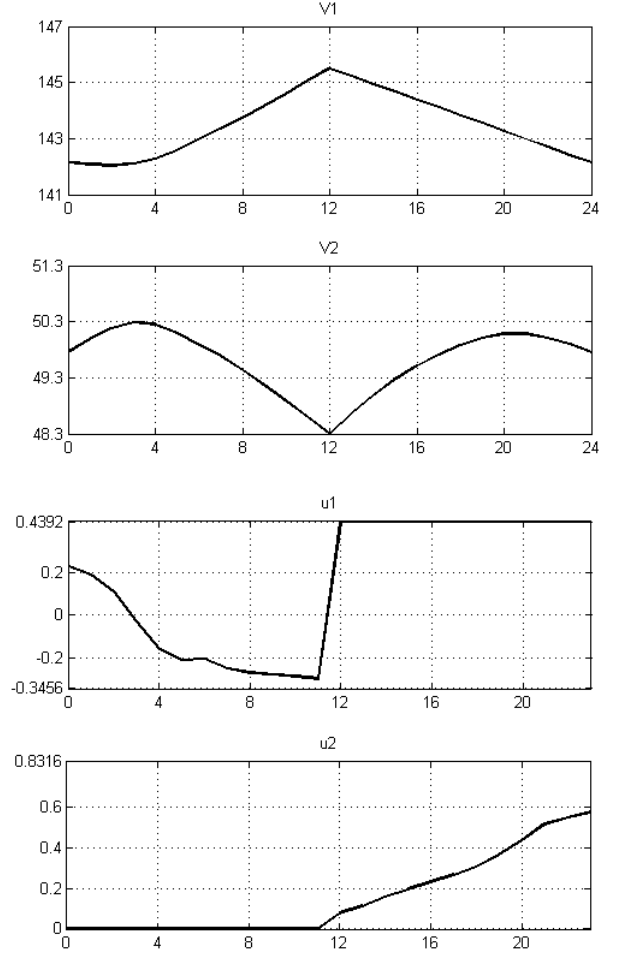


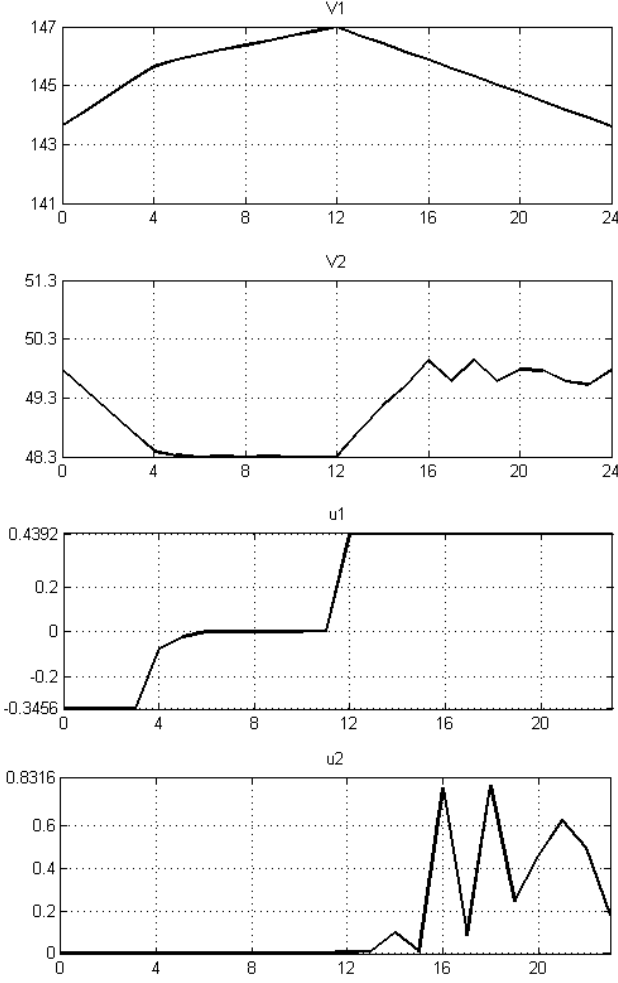
Fig. 6: Approximate solution

The following table summarizes the steps involved in the two approaches and shows costs and execution times.

	1st approach	2nd approach
	• Chen - Burer Algorithm (directly)	• PER • Chen - Burer Algorithm • QuadProg • Local optimization
<b>Cost</b>	-308.9	-308.6
<b>Total time execution</b>	24 hours	1.48 min

Table 1: Comparison of methods





**Fig. 7:** Final results with second approach

The timings were obtained in the same computer, using Matlab R 2011b, under a 64-bit windows 7 Home Premium. However, as we can see in the table, execution times are quite different. The costs associated with the two approaches are almost the same.

Comparing figures 2 and 7, we realize that when we apply directly the Chen-Burer algorithm or the procedure described in section 3.2.1 with PER method, the trajectories are similar, although  $V_2(t)$  is quite different in the second half part of the interval. Let us see that such irregular behavior can be justified by the existence of an infinity of solutions to the original problem defined by (1), (2) and (5).

The fact that  $u_1(t) = u_1^M$ ,  $t \in [T/2, T]$  and  $V_1(T/2) = V_1^M$ ,  $V_2(T/2) = V_2^m$ , allows us to write,

$$\begin{aligned} V_1(T) &= V_1(T/2) + \int_{T/2}^T (A - u_1^M) dt, \\ &= V_1^M + (A - u_1^M) \frac{T}{2} \\ V_2(T) &= V_2(T/2) + \int_{T/2}^T (u_1^M - u_2(t)) dt, \\ &= V_2^m + u_1^M \frac{T}{2} - \int_{T/2}^T u_2(t) dt. \end{aligned}$$

Observe that  $V_2(0) = V_2(T)$ , so the cost function (5) depends only on  $V_2(T/2)$  and  $V_2(T)$ , in what respects the variable  $V_2$ .

Maintaining the values of  $V_2(T/2)$  and  $V_2(T)$  let us check the existence of more than one admissible control function  $u_2(\cdot)$  on  $[T/2, T]$ , keeping  $V_2(t)$  also admissible. Take  $u_2(t) = \text{const} = \bar{u}_2$ ,  $t \in [T/2, T]$ , for some  $\bar{u}_2$ . To preserve the value of  $V_2(T)$ , we must have

$$\bar{u}_2 = \frac{2}{T} \left( -V_2(T) + V_2^m + \frac{T}{2} u_1^M \right).$$

For our particular data the above value for  $\bar{u}_2$  is in the interval  $[u_2^m, u_2^M]$  and the resulting  $V_2(t)$  is admissible. This leads to other optimal solution for the problem, different from that presented in figures. The solution for the problem is not unique.

Furthermore, we can show that there is an infinity of solutions. For the above  $\bar{u}_2$ , taking  $u_2(t) = \bar{u}_2$  we have

$$\int_{T/2}^T u_2(t) dt = \bar{u}_2 \frac{T}{2}.$$

Define now a piecewise constant function  $u_2(t)$  in the following way

$$u_2(t) = \begin{cases} w_1, & t \in [T/2, \theta[ \\ w_2, & t \in [\theta, T] \end{cases}$$

with  $\theta$ ,  $w_1$ ,  $w_2$  constants. Then

$$\begin{aligned} \int_{T/2}^T u_2(t) dt &= \int_{T/2}^{\theta} w_1 dt + \int_{\theta}^T w_2 dt \\ &= w_1(\theta - T/2) + w_2(T - \theta). \end{aligned}$$

If

$$\theta = \frac{T(\bar{u}_2 + w_1 - 2w_2)}{2(w_1 - w_2)} \quad (16)$$

we still have  $\int_{T/2}^T u_2(t) dt = \bar{u}_2 \frac{T}{2}$ . There are an infinity of

solutions  $(\theta, w_1, w_2)$  with  $\theta \in ]T/2, T[$ ,  $w_1, w_2 \in [u_2^m, u_2^M]$ , satisfying (16) and keeping  $V_2(t)$  admissible.

The cost function has the same value for functions  $u_2(t)$  defined as above. This explains the possible different behaviors for controls and trajectories, on  $[T/2, T]$ , in figures 2 and 7.

## Acknowledgement

This work has been partially supported by the EU 7th Framework Prog. [FP7-PEOPLE-2010-ITN] grant agr. 64735-SADCO, FCT project PTDC/EEA-CRO/116014/2009 and project FCOMP-01-0124-FEDER-028894, Ref.FCT PTDC/EEI-AUT/1450/2012.

## References

- [1] V.A Bushenkov, "An iteration method of constructing orthogonal projections of convex polyhedral sets," U.S.S.R. Comput. Maths. Math. Phys. **20(3)**, 1-5, (1995)
- [2] S.N Tchernikov, "Lineare Ungleichungen", Deutscher Verlag der Wissenschaften, Berlin, (1971)
- [3] O.L Chernykh, "Construction of the convex hull of a finite set of points when the computations are approximate", U.S.S.R. Comput. Maths. Math.Phys. **28(5)**, 71-77, (1988)
- [4] J Chen, S Burer, "Globally solving nonconvex quadratic programming problems via completely positive programming", Math. Prog. Comp., **4(1)**, 33-52, (2012)
- [5] V.A Bushenkov, M.M.A Ferreira, A.F Ribeiro, G.V Smirnov, "Numerical approach to a problem of hydroelectric resources management", AIP Conference Proceedings **1558**, 630-633 (2013)
- [6] A.V Lotov, V.A Bushenkov, G.K Kamenev, "Interactive Decision Maps: Approximation and Visualization of Pareto Frontier", Springer (2004)
- [7] G Smirnov, V Bushenkov, "Curso de Optimiza  o: Programaa  o Matem  tica,C  culo das Varia   es, Controlo   ptimo", Escolar Editora (in Portuguese), ()2005)
- [8] A.V Lotov, A.I Pospelov, "The modified method of refined bounds for polyhedral approximation of convex polytopes", Computational Mathematics and Mathematical Physics, **48(6)**, 933-941 (2008)
- [9] J.W Labadie, "Optimal Operation of multireservoir systems: state-of-the-art review", J. Water Resour. Plng. and Mgmt., **130(2)**, 93-111 (2004)
- [10] S Burer, "Optimizing a polyhedral-semidefinite relaxation of completely positive programs", Mathematical Programming Computation, **2(1)**, 1-19 (2010)
- [11] S Burer, "On the copositive representation of binary and continuous nonconvex quadratic programs", Mathematical Programmin, **120(2)**, 479-495 (2009)
- [12] S Burer, D Vandenbussche, "A finite branch-and-bound algorithm for nonconvex quadratic programming via semidefinite relaxations", Mathematical Programmin, **113(2)**, 259-282 (2008)
- [13] S.J.P Mariano, J.P.S Catalao, V.M.F Mendes, L.A.F.M Ferreira, "Profit-Based Short-Term Hydro Scheduling considering Head-Dependent Power Generation", Power Tech, 2007 IEEE Lausanne, 1362-1367 (2007)