Constructive heuristics and local search for a large-scale energy management problem

Mauro Dell'Amico José Carlos Díaz Díaz
Team S4

Interuniversity Consortium for Optimization and Operations Research (ICOOR)
University of Modena and Reggio Emilia

July 14, 2010

Outline

- Introduction
- Preprocessing phase
- Constructive heuristics
- Local search
- Mathematical Model
- Computational results

Introduction

The general aim is to plan the production of electrical energy using two sets of *powerplants* (*Type 1* and *Type 2*), along a given time horizon of T *timesteps*, grouped into *weeks* of identical length.

Introduction

The general aim is to plan the production of electrical energy using two sets of *powerplants* (*Type 1* and *Type 2*), along a given time horizon of T *timesteps*, grouped into *weeks* of identical length.

The exact demand of energy for each *timestep* is unknown, so it is represented by means of S *scenarios* each described by a demand profile (i.e., a vector $DEM^{ts}, t = 0, \dots, T-1$, for each scenario s).

Introduction

The general aim is to plan the production of electrical energy using two sets of *powerplants* (*Type 1* and *Type 2*), along a given time horizon of T *timesteps*, grouped into *weeks* of identical length.

The exact demand of energy for each *timestep* is unknown, so it is represented by means of S *scenarios* each described by a demand profile (i.e., a vector DEM^{ts}, $t = 0, \ldots, T-1$, for each scenario s).

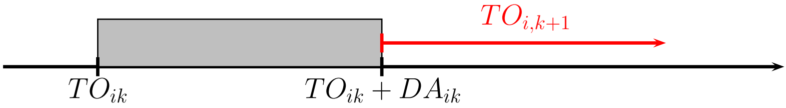
Each powerplant of *type 1* may have a minimum and a maximum power bound for each *timestep* and *scenario*, while the prowerplants of *type 2* have several kind of constraints.

Preprocessing phase

- Strength and propagate the problem constraints,
- Reduce the size and complexity of the overall problem,
- Reduce the time windows (constraint CT13),
- Lower bounds of the time windows (if not given) are fixed through six criteria.

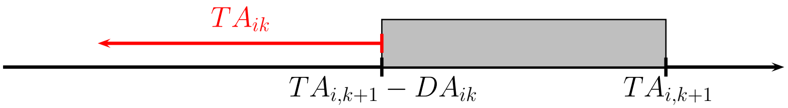
Criteria with DA

C1 given the starting time TO_{ik} and the duration of the outage of this cycle k of powerplant i (DA_{ik}), the starting time of the next cycle $TO_{i,k+1} \geq TO_{ik} + DA_{ik}$.



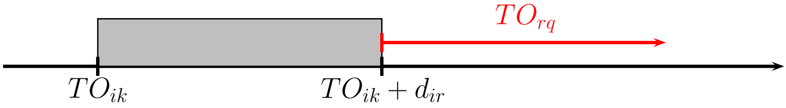
Criteria with *DA*

C2 **if** ($TA_{i,k+1} \neq -1$) **then** $TA_{ik} \leq TA_{i,k+1} - DA_{ik}$.
Strengthen the ending time of the time windows on a cycle k, given a defined ending time of cycle $k+1$.



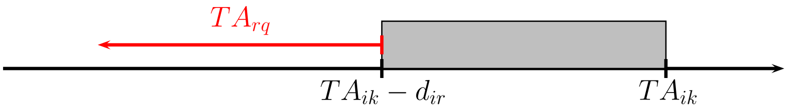
Criteria with CT14 – CT18

C3 **if** ($TO_{ik} + DA_{ik} < TO_{rq} + DA_{rq}$ **and** $TO_{ik} \neq -1$) **then** the starting time of powerplant r of the cycle q is $TO_{rq} \geq TO_{ik} + d_{ir}$, where $d_{ir} = \max\{Se^{14}_{ir} + DA_{ik}, Se^{15}_{ir} + DA_{ik}, Se^{16}_{ir}, Se^{17}_{ir} + DA_{ik} - DA_{rq}, Se^{18}_{ir} - DA_{rq}\}$ is the maximum spacing between the powerplants i and r given by constraints CT14, CT15, CT16, CT17 and CT18.



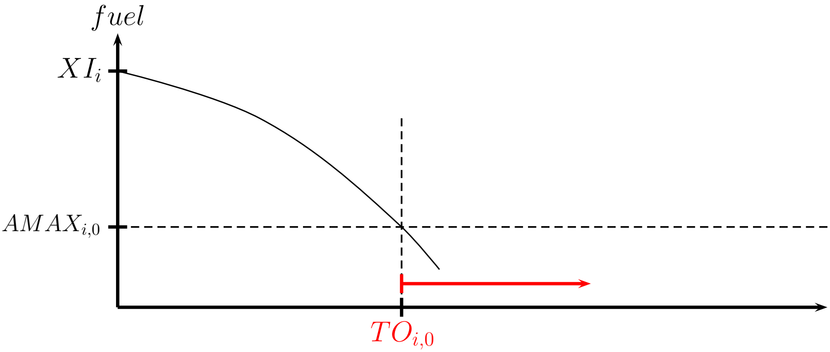
Criteria with CT14 – CT18

C4 **if** ($TA_{ik} + DA_{ik} > TA_{rq} + DA_{rq}$ **and** $TA_{ik} \neq -1$) **then**
the ending time of powerplant r on cycle q is
$TA_{rq} \leq TA_{ik} - d_{ir}$, where
$d_{ir} = \max\{Se^{14}_{ir} + DA_{rq}, Se^{15}_{ir} + DA_{rq}, Se^{16}_{ir}, Se^{17}_{ir} - DA_{ik} + DA_{rq}, Se^{18}_{ir} + DA_{ik}\}$
is the maximum spacing between the powerplants i and r given by constraints CT14, CT15, CT16, CT17 and CT18.



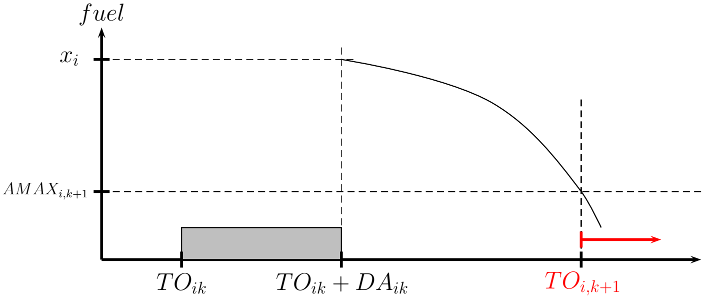
Criteria with CT11

C5 Using the starting fuel XI_i at powerplant i, we can compute the minimum timestep in which the *fuel* is smaller than or equal to the limit $AMAX_{i,0}$. This timestep is a lower bound for $TO_{i,0}$.



Criteria with CT11

C6 By using the smallest amount of fuel x_i in the timestep $TO_{ik} + DA_{ik}$, we can compute the minimum timestep in which the *fuel* is smaller than or equal to the limit $AMAX_{i,k+1}$. This timestep is a lower bound for $TO_{i,k+1}$.



Pseudocode

1: **call** C1(), C2();
2: **call** C3(), C4() **if** any update occurs **then go to** step 1;
3: **call** C5(), C6() **if** any update occurs **then go to** step 1;

Example. Instance data0.txt

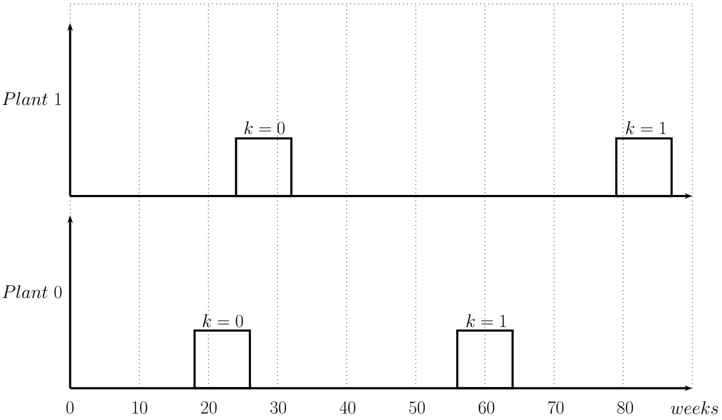


Figure: Time windows of CT13

Example. Instance data0.txt

	CT13		**CT14**
$DA_{ik} = \begin{pmatrix} 5 & 8 \\ 9 & 6 \end{pmatrix}$	$TO_{ik} = \begin{pmatrix} 18 & 56 \\ 24 & 79 \end{pmatrix}$	$TA_{ik} = \begin{pmatrix} 26 & 64 \\ 32 & 87 \end{pmatrix}$	$Se_{A \times A} = 6, A = \{0, 1\}$

Example. Instance data0.txt

	CT13		CT14
$DA_{ik} = \begin{pmatrix} 5 & 8 \\ 9 & 6 \end{pmatrix}$	$TO_{ik} = \begin{pmatrix} 18 & 56 \\ 24 & 79 \end{pmatrix}$	$TA_{ik} = \begin{pmatrix} 26 & 64 \\ 32 & 87 \end{pmatrix}$	$Se_{A \times A} = 6, A = \{0, 1\}$

C1:

i	k	$TO_{ik} + DA_{ik}$	$TO_{i,k+1}$	$TO_{ik} + DA_{ik} \leq TO_{i,k+1}$
0	0	$18 + 5$	56	true
1	0	$24 + 9$	79	true

Example. Instance data0.txt

	CT13		CT14
$DA_{ik} = \begin{pmatrix} 5 & 8 \\ 9 & 6 \end{pmatrix}$	$TO_{ik} = \begin{pmatrix} 18 & 56 \\ 24 & 79 \end{pmatrix}$	$TA_{ik} = \begin{pmatrix} 26 & 64 \\ 32 & 87 \end{pmatrix}$	$Se_{A \times A} = 6, A = \{0, 1\}$

C1:

i	k	$TO_{ik} + DA_{ik}$	$TO_{i,k+1}$	$TO_{ik} + DA_{ik} \leq TO_{i,k+1}$
0	0	$18 + 5$	56	true
1	0	$24 + 9$	79	true

C2:

i	k	TA_{ik}	$TA_{i,k+1} - DA_{ik}$	$TA_{ik} \leq TA_{i,k+1} - DA_{ik}$
0	1	26	$64 - 5$	true
1	1	32	$87 - 9$	true

Example. Instance data0.txt

	CT13		CT14
$DA_{ik} = \begin{pmatrix} 5 & 8 \\ 9 & 6 \end{pmatrix}$	$TO_{ik} = \begin{pmatrix} 18 & 56 \\ 24 & 79 \end{pmatrix}$	$TA_{ik} = \begin{pmatrix} 26 & 64 \\ 32 & 87 \end{pmatrix}$	$Se_{A \times A} = 6, A = \{0, 1\}$

C3:

i	k	r	q	$TO_{ik} + d_{ir}$	TO_{rq}	$TO_{ik} + d_{ir} \leq TO_{rq}$
0	0	0	1	$18 + (6 + 5)$	56	true
0	0	1	0	$18 + (6 + 5)$	24	false
0	0	1	1	$18 + (6 + 5)$	79	true
1	0	0	1	$24 + (6 + 9)$	56	true
1	0	1	1	$24 + (6 + 9)$	79	true

$TO_{1,0} = 29$

Example. Instance data0.txt

	CT13		CT14
$DA_{ik} = \begin{pmatrix} 5 & 8 \\ 9 & 6 \end{pmatrix}$	$TO_{ik} = \begin{pmatrix} 18 & 56 \\ 24 & 79 \end{pmatrix}$	$TA_{ik} = \begin{pmatrix} 26 & 64 \\ 32 & 87 \end{pmatrix}$	$Se_{A \times A} = 6, A = \{0, 1\}$

	i	k	r	q	TA_{rq}	$TA_{ik} - d_{ir}$	$TA_{rq} \leq TA_{ik} - d_{ir}$	
	0	1	0	0	26	$56 - (6+5)$	true	
	0	1	1	0	24	$56 - (6+6)$	true	
C4:	0	1	0	0	26	$32 - (6+5)$	false	$TA_{0,0} = 21$
	1	1	0	0	26	$87 - (6+5)$	true	
	1	1	0	1	56	$87 - (6+5)$	true	
	1	1	1	0	24	$87 - (6+5)$	true	

Example. Instance data0.txt

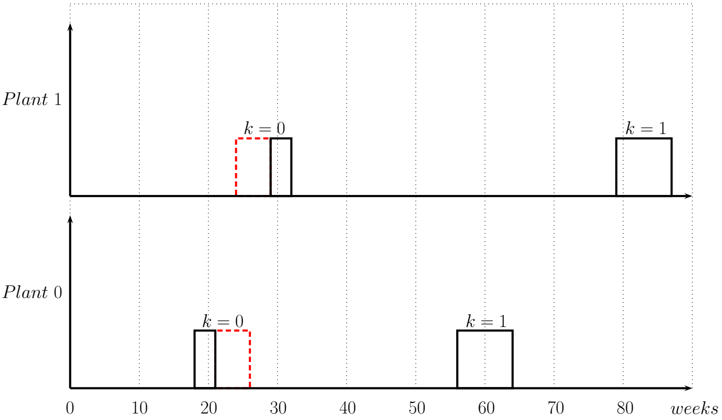


Figure: Updated time windows

Constructive heuristics

We have designed two greedy algorithms to find a starting feasible solution:

Greedy$_{TO}$() Select the outages by increasing time, assign each of them as close as possible to the starting times (TO) of the time windows.

Constructive heuristics

We have designed two greedy algorithms to find a starting feasible solution:

Greedy$_{TO}$() Select the outages by increasing time, assign each of them as close as possible to the starting times (TO) of the time windows.

Greedy$_{TA}$() Select the outages by decreasing time, assign each of them as close as possible to the starting times (TA) of the time windows.

Constructive heuristics

The idea is to fix the outages of the Type 2 powerplants, providing that a feasible power and fuel assignment exists for all scenarios. Given this assignment we complete the solution using the Type 1 powerplants to satisfy the demands. For this scope we use three algorithms:

***FindOutage*()** Find a feasible outage ha_{ik} (CT14 – CT18).

Constructive heuristics

The idea is to fix the outages of the Type 2 powerplants, providing that a feasible power and fuel assignment exists for all scenarios. Given this assignment we complete the solution using the Type 1 powerplants to satisfy the demands. For this scope we use three algorithms:

FindOutage() Find a feasible outage ha_{ik} (CT14 – CT18).

ConstCheck() Check the feasibility of the outage by constraints CT19 – CT21 and CT11. If necessary update the time windows (CT13).

Constructive heuristics

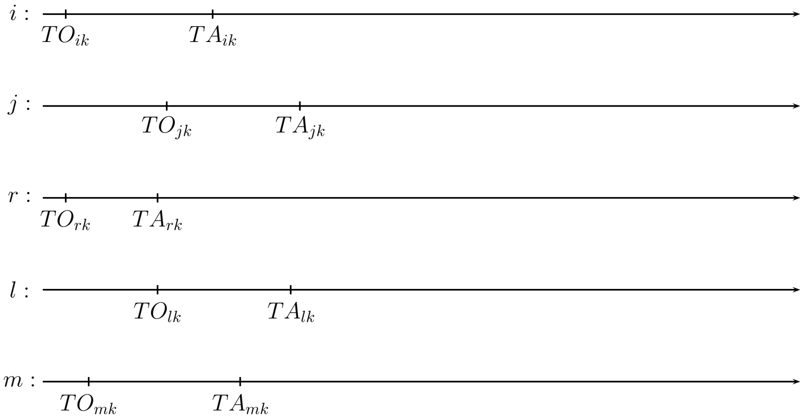
The idea is to fix the outages of the Type 2 powerplants, providing that a feasible power and fuel assignment exists for all scenarios. Given this assignment we complete the solution using the Type 1 powerplants to satisfy the demands. For this scope we use three algorithms:

***FindOutage*()** Find a feasible outage ha_{ik} (CT14 – CT18).

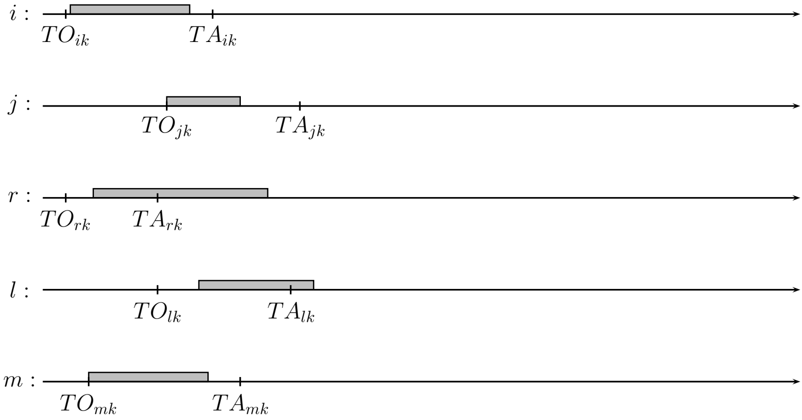
***ConstCheck*()** Check the feasibility of the outage by constraints CT19 – CT21 and CT11. If necessary update the time windows (CT13).

***PlanPower*()** Plan the power and fuel for each timestep and scenario, given the outage ha_{ik} and the refueling r_{ik}.

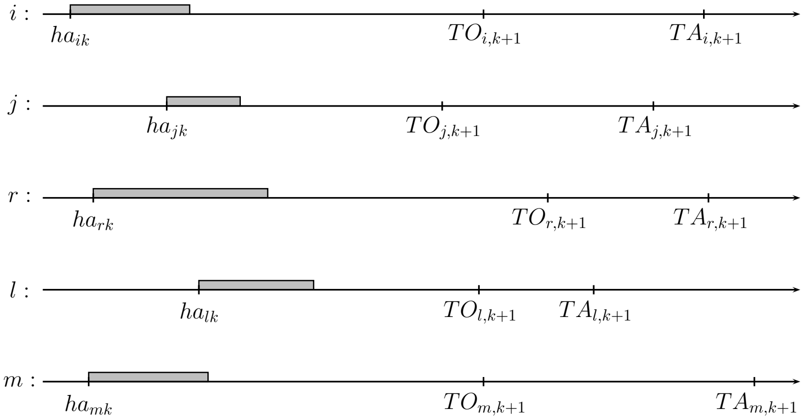
FindOutage() algorithm in Greedy$_{TO}$()



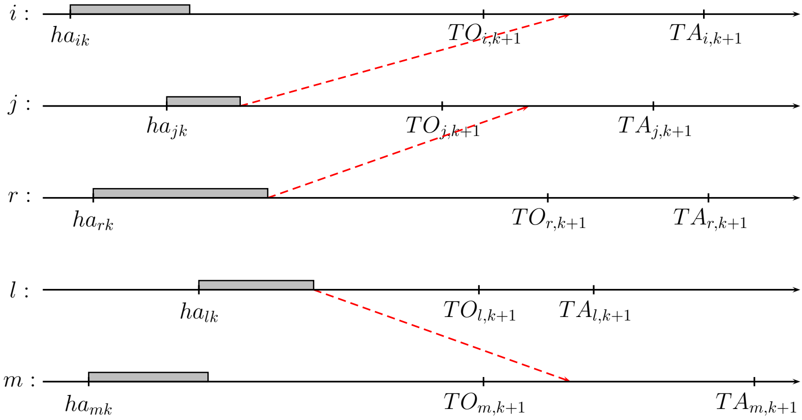
FindOutage() algorithm in Greedy$_{TO}$()



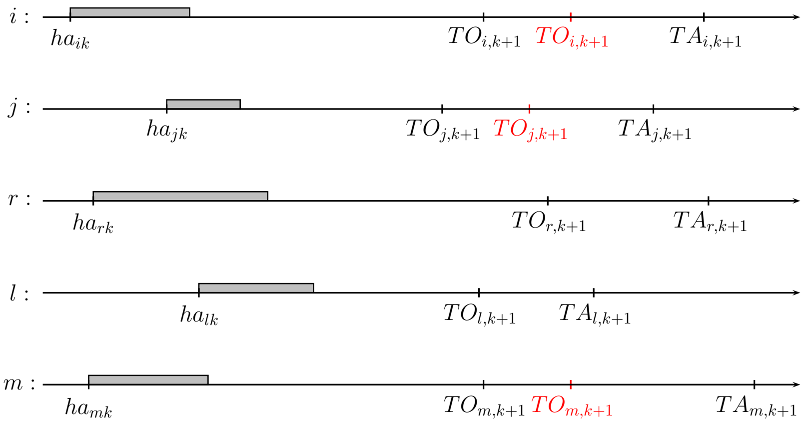
FindOutage() algorithm in Greedy$_{TO}$()



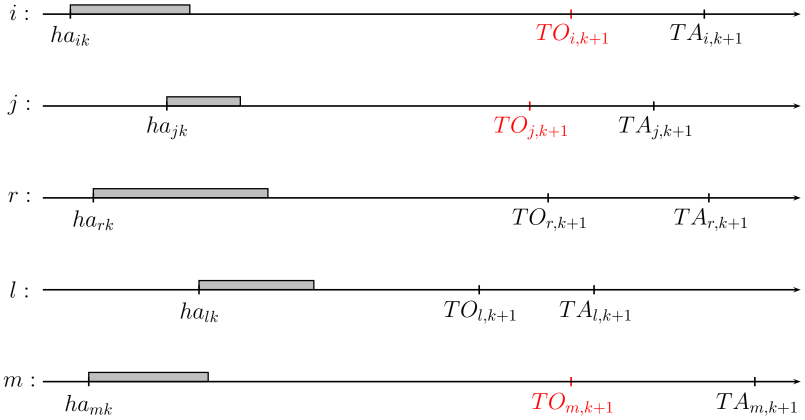
FindOutage() algorithm in Greedy$_{TO}$()



FindOutage() algorithm in Greedy$_{TO}$()



FindOutage() algorithm in Greedy$_{TO}$()



Pseudocode

$ha_{ik} = -1,\ r_{ik} = RMIN_{ik}, \quad \forall i, k;$
repeat
    call *FindOutage*();
    status := *ConstCheck*(CT11, CT19, CT20, CT21);
until *status* = feasible
call *PlanPower*();

Local search

We have designed two local search procedures to improve the solutions obtained with the greedy algorithms:

$\text{LS}_r()$ Try to improve the power assigned to the *Type* 2 plants by giving them a larger fuel and check for feasibility constraint CT11. If necessary, reduce the refueling r_{ik} and update the bounds *AMAX* and *SMAX*.

Local search

We have designed two local search procedures to improve the solutions obtained with the greedy algorithms:

$LS_r()$ Try to improve the power assigned to the *Type* 2 plants by giving them a larger fuel and check for feasibility constraint CT11. If necessary, reduce the refueling r_{ik} and update the bounds *AMAX* and *SMAX*.

$LS_{ha}()$ Try to improve the solution by changing the outage dates. Start from the beginning (or last) cycle and we consider the outages of all powerplants. Using a scoring function we select a plant i and increase (or decrease) the starting of its outage in the current cycle.

Local search: $LS_r()$

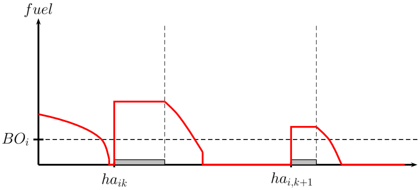


Figure: Initial solution.

Local search: $\text{LS}_r()$

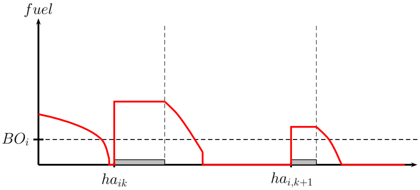


Figure: Initial solution.

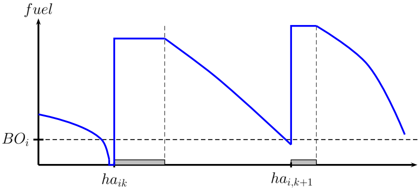


Figure: Improved solution with $\text{LS}_r()$.

Mathematical model

Starting from a feasible outage assignment, two mathematical models have been designed and implemented to define the best fuel and power assignment for Type 2 powerplants:

Model 1 decompose the problem where the time granularity corresponds to the week.

Mathematical model

Starting from a feasible outage assignment, two mathematical models have been designed and implemented to define the best fuel and power assignment for Type 2 powerplants:

Model 1 decompose the problem where the time granularity corresponds to the week.

Model 2 plans the timesteps of a given week w, by taking into account the higher level planning of the first model.

Model 2

$$\max \sum_{i \in I, t \in T, s \in S} p_{its} - M \sum_{i \in I, s \in S} (lp_{is} + up_{is} + lx_{is} + ux_{is}) \quad (1)$$

$$\sum_{t \in w} p_{its} + lp_{is} - up_{is} = \bar{p}_{iws} \quad \forall i, s, \quad (2)$$

$$p_{its} = profile(i, t, s) \quad \forall i, t, s, \quad (3)$$

$$x_{i,t+1,s} = x_{its} - p_{its} D^t \quad \forall i, k, s, t \in w, \quad (4)$$

$$x_{i,t+1,s} = \frac{Q_{ik} - 1}{Q_{ik}} (x_{its} - BO_{i,k-1}) + + r_{ik} + BO_{ik} \quad \forall i, k, s, t, \quad (5)$$

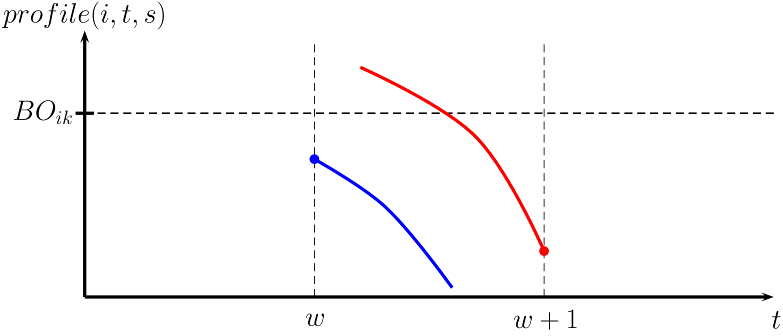
$$x_{its} + lx_{is} - ux_{is} = \bar{x}_{iws} \quad \forall i, s, w : t = \text{ first } w \quad (6)$$

$$0 \leq x_{its} \leq AMAX_{ik} \quad \forall i, k, s, t = \text{first } ea(i, k),$$

$$x_{i,t+1,s} \leq SMAX_{ik} \quad (7)$$

$$p_{its}, x_{its}, lp_{is}, up_{is}, lx_{is}, ux_{is} \geq 0 \quad \forall i, t, s. \quad (8)$$

Function *profile*(i, t, s)



Computational results

We implemented all algorithms in C++ and run them on a Intel Xeon, 2.40GHz, 8MB Cache with 6GB of RAM and running under o.s. Linux Ubuntu 10.04.

Table: Results of our algorithms.

Data A			*Data B*		
name	*total cost*	*t (sec)*	*name*	*total cost*	*t (sec)*
data0	8.735435262138E12	1800	data6	8.9659069433E10	3600
data1	1.69625914405E11	1800	data7	8.6134819374E10	3600
data2	1.46208292628E11	1800	data8	9.4736433662E10	3600
data3	1.54653401636E11	1800	data9	1.02833931055E11	3600
data4	1.12301533269E11	1800	data10	8.4353400440E10	3600
data5	1.28192364678E11	1800			

Conclusions

Thank you for your attention!!