DCA for minimizing the cost and tardiness of preventive maintenance tasks under real-time allocation constraints

September 15, 2013

Outline

Problem

- We consider a system composed of N independent machines that may request renewable maintenance jobs by q parallel processors.
- The number of processors is limited compared to the number of machines in the system, $q << N$

Problem

- We consider the time horizon $[0, T]$
- At the initial time, all machines are brand-new.
- The processing time p_i is specific and given for each machine.

Problem

- On each machine, except the first job, the release date and the due date of the other jobs are computed by the end of the precedent one.
- We note that c_{i,k_i}, r_{i,k_i}, d_{i,k_i} are respectively the completion date, the release date, the due date of the k_i^{th} job on the i^{th} machine.

$$r_{i,k_i} = c_{i,k_i-1} + \rho_i$$

$$d_{i,k_i} = c_{i,k_i-1} + \delta_i$$

where $\rho_i < \delta_i$ and they are given for each machine.

Problem

The total of makespan and the tardiness of jobs on the time horizon T is expressed by:

$$C_T = \sum_{i=1}^{N} \Big(\sum_{k_i / r_{i,k_i} \in T} \big(W_0(c_{i,k_i} - r_{i,k_i}) + W_1 \max(0, c_{i,k_i} - d_{i,k_i}) \big) \Big)$$

where W_0 and W_1 represent respectively the costs per time period of the jobs without and with tardiness.

The aim is to minimize the total of makespan and tardiness on the time horizon. It means that we have to minimize C_T

Mathematical formulation

We divide the time horizon into H intervals. We introduce the decision variables $x_{i,t}$ defined as: $x_{i,t} = 1$ if i^{th} machine maintenance takes place at the time t, 0 otherwise.The required constraints on the number q of available processors is then expressed as

$$\sum_{i=1}^{N} x_{i,t} \leq q \quad \forall t = 1, 2, .., H. \tag{1}$$

Mathematical formulation

The constraints ensuring that the duration between two consecutive maintenance jobs on the i^{th} machine is greater than ρ_i can be written as

$$x_{i,t} + x_{i,t+1} + \ldots + x_{i,t+\rho_i+p_i-1} \leq p_i \quad \forall\ t = 1, 2, \ldots, H - \rho_i - p_i + 1. \tag{2}$$

Mathematical formulation

Since every machine is brand-new at the beginning of the considered horizon, we know that

$$x_{i,t} = 0 \quad \forall i, \; \forall t < \rho_i. \tag{3}$$

For assuring that the job processing time of the i^{th} machine is p_i, we need

$$x_{i,H} + \sum_{t=1}^{H-1} |x_{i,t+1} - x_{i,t}| = \frac{2}{p_i} \sum_{t=1}^{H} x_{i,t}.$$

This constraint is not linear. We replace it by the set of linear constraints:

Mathematical formulation

$$\sum_{t=1}^{H} z_{i,t} = \frac{2}{p_i} \sum_{t=1}^{H} x_{i,t} \quad \forall i = 1, 2, ..., N; \tag{4}$$

$$x_{i,t+1} - x_{i,t} \leq z_{i,t} \quad \forall i = 1, 2, .., N \; \forall t = 1, 2, .., H - 1; \tag{5}$$

$$x_{i,t} - x_{i,t+1} \leq z_{i,t} \quad \forall i = 1, 2, .., N \; \forall t = 1, 2, .., H - 1; \tag{6}$$

$$z_{i,H} = x_{i,H} \quad \forall i = 1, 2, .., N; \tag{7}$$

$$0 \leq z_{i,t} \leq 1 \quad \forall i, \; \forall t = 1, 2, .., H. \tag{8}$$

Mathematical formulation

We now define the objective function of our optimization problem.

It is easy to see that $\frac{1}{p_i}\sum_{i=1}^{H} x_{i,t}$ is the maintenance number of i^{th} machine in horizon $[0, H]$, the flow-time of i^{th} machine in horizon $[0, H]$ is computed as follows:

$$\sum_{k_i/r_{i,k_i}\in H}(c_{i,k_i} - r_{i,k_i}) = H - (1 + \frac{1}{p_i}\sum_{i=1}^{H} x_{i,t}).\rho_i.$$

Mathematical formulation

For computing the tardiness, we consider two consecutive maintenances (see Figure).

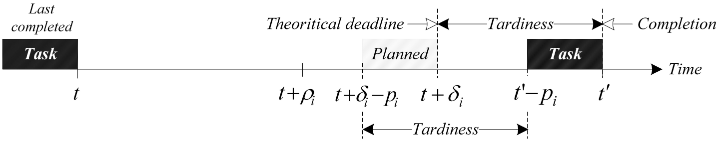


Figure:

In this figure, t represents the completion date of the precedent job. $t + \delta_i$, t', $t' - p_i$ are, respectively, the due date, the completion date, the beginning date of the next job. The tardiness is $t' - t - \delta_i$.

Mathematical formulation

By considering the variables $w_{i,t}$ satisfying

$$0 \leq w_{i,t} \leq 1$$

and

$$1 - (x_{i,t} + x_{i,t+1} + ... + x_{i,t+\delta_i - p_i - 1}) \leq w_{i,t}.$$

The tardiness is the number of $w_{i,t}$ which are equal to 1. The tardiness cost is so expressed by $W_1. \sum_{t=1}^{H-\delta_i+p_i+1} w_{i,t}$.

Mathematical formulation

The total cost on the i^{th} machine is therefore expressed as

$$C = W_0.H - W_0(1 + \frac{1}{p_i}\sum_{i=1}^{H} x_{i,t}).\rho_i + W_1.\sum_{t=1}^{H-\delta_i+p_i+1} w_{i,t}$$

.

Mathematical formulation

Finally, the optimization model of our problem can be written in the form

$$(P) \qquad \min \sum_{i=1}^{N} \left\{ W_0.(H-\rho_i) - W_0.\frac{\rho_i}{p_i} \sum_{t=1}^{H} x_{i,t} + W_1. \sum_{t=1}^{H-\delta_i+p_i+1} w_{i,t} \right\},$$

subject to the constraints (1-8) and

$$x_{i,t} + x_{i,t+1} + \dots + x_{i,t+\delta_i-p_i-1} + w_{i,t} \geq 1 \quad \forall i,\ \forall t = 1,2,..,H-\delta_i+p_i+1. \tag{9}$$

$$0 \leq w_{i,t} \leq 1 \quad \forall i,\ \forall t = 1,2,...,H-\delta_i+p_i+1. \tag{10}$$

$$x_{i,t} \in \{0,1\} \quad \forall i,\ \forall t = 1,2,..,H. \tag{11}$$

How to solve the problem (P)?

Method based on DC Programming and DCA.

DC programming

DC = difference of convex functions;DCA = DC Algorithm

A DC programming takes the form

$$(P) \quad \alpha = \inf\{f(x) := g(x) - h(x) : x \in \mathbb{R}^n\}$$

where g and h are l.s.c(lower semi continuous) proper convex functions on $\mathbb{R}^n$

The dual problem of (P):

$$(D) \quad \alpha = \inf\{h^*(y) - g^*(y) : y \in \mathbb{R}^n\}$$

where

$$g^*(y) = \sup\{\langle x, y\rangle - g(x) : x \in \mathbb{R}^n\}$$

(the conjugate function)

DC Algorithm

About DCA and its applications (http://lita.sciences.univ-metz.fr/ lethi/DCA.html)

Introduced in 1985 by PHAM DINH Tao

Developped since 1994 by LE THI Hoai An and PHAM DINH Tao

Based on local optimality and the duality in DC

DCA to solve DC programming problem

- Initialization. Choose $x^0 \in \mathbb{R}^n$; set $k = 0$.
- Step 1. Compute a subgradient $y^k \in \partial h(x^k)$.
- Step 2. Compute

$$x^{k+1} \in \mathrm{Argmin}\{g(x) - \langle y^k, x\rangle : \ x \in \mathbb{R}^n\}$$

- Step 3. If the STOPPING CRITERIA is satisfied, STOP; Otherwise, set $k = k + 1$, go to Step 1.

Penalty Theorem

Theorem(H.A. LE THI, T.PHAM DINH, and M.LE DUNG)
Let K be a nonempty bounded polyhedral convex set, f be a finite concave function on K and p be a finite nonnegative concave function on K. Then there exists $\eta_0 \geq 0$ such that for $\eta > \eta_0$ the following problems have the same optimal value and the same solution set

$$(P_t) \qquad \alpha(t) = \min\left\{f(x) + \eta.p(x) : x \in K\right\}$$

$$(P) \qquad \alpha = \min\left\{f(x) : x \in K, p(x) \leq 0\right\}$$

Furthermore

- If the vertex set of K, denoted by $V(K)$, is contained in $x \in K : p(x) \leq 0$, then $\eta_0 = 0$.
- If $p(x) > 0$ for some x in $V(K)$, then $\eta_0 = \min\left\{\frac{f(x)-\alpha(0)}{S_0} : x \in K, p(x) \leq 0\right\}$, where $S_0 = \min\left\{p(x) : x \in V(K), p(x) > 0\right\} > 0$.

DCA applied to the problem (P)

Let L be the number of variables of the problem (P), $L = 2NH + \sum_{i=1}^{N}(H - \delta_i + p_i + 1)$. Denote by S the feasible set of (P) and let the linear relaxation domain of S be
$K := \{(x, z, w) \in \mathbb{R}^L$ satisfy (1-10) and $x_{i,t} \in [0, 1]\}$
We consider the function $p : \mathbb{R}^L \to \mathbb{R}$ defined by
$p(x, z, w) = \sum_{i=1}^{N} \sum_{t=1}^{H} x_{i,t}(1 - x_{i,t})$.

DCA applied to the problem (P)

By using penalty theorem, we obtain, for the sufficiently large number η, the equivalent concave minimization problem to (P)
$\min \left\{ \sum_{i=1}^{N} \left\{ W_0(H-\rho_i) - W_0 \frac{\rho_i}{p_i} \sum_{t=1}^{H} x_{i,t} + W_1 . \sum_{t=1}^{H-\delta_i+p_i+1} w_{i,t} \right\} + \eta . p(x,z,w) : (x,z,w) \in K \right\}$. This is a DC problem whose feasible set is K and the objective function is

$$f_\eta(x,z,w) = g(x,z,w) - h(x,z,w) \text{ , where} \tag{12}$$

$$g(x,z,w) := \chi_K(x,z,w);$$

$$h(x,z,w) := -\sum_{i=1}^{N} \left\{ W_0(H-\rho_i) - W_0 . \frac{\rho_i}{p_i} \sum_{t=1}^{H} x_{i,t} + W_1 . \sum_{t=1}^{H-\delta_i+p_i+1} w_{i,t} \right\} + \eta . \sum_{i=1}^{N} \sum_{t=1}^{H} x_{i,t}(x_{i,t}-1).$$

DCA applied to the problem (P)

DCA applied to the DC program (12) consists of computing, at each iteration ℓ, the two sequences $\left\{u^l\right\}$ and $\left\{\varsigma^\ell = (x^\ell, z^\ell, w^\ell)\right\}$. $(x^{\ell+1}, z^{\ell+1}, w^{\ell+1})$ is obtained by solving the next linear program

$$\min\{-\langle (x,z,w), u^\ell\rangle : (x,z,w) \in K\}. \tag{13}$$

From the definition of h, $u^\ell \in \partial h(x^\ell, z^\ell, w^\ell)$ is computed as

$$\begin{cases} u^\ell_{H.(i-1)+t} = 2.\eta x^\ell_{i,t} + W_0.\frac{\rho_i}{p_i} \quad \forall i = 1,2,..,N, \forall t = 1,2,...,H, \\ u^\ell_j = 0 \quad \forall j = NH, NH+1, ..., 2NH. \\ u^\ell_j = -W_1 \quad \text{otherwise.} \end{cases} \tag{14}$$

DCA applied to the problem (P)

The DCA applied to (12) can be described as follows.

Initialization

Let ϵ be a sufficiently small positive number. Set $\ell = 0$ and $(x^0, z^0, w^0) \in \mathbb{R}^L$.

Repeat

Calculate $u^\ell \in \partial h(x^\ell, z^\ell, w^\ell)$ via (14).

Solve the linear problem (13) to obtain $(x^{\ell+1}, z^{\ell+1}, w^{\ell+1})$.

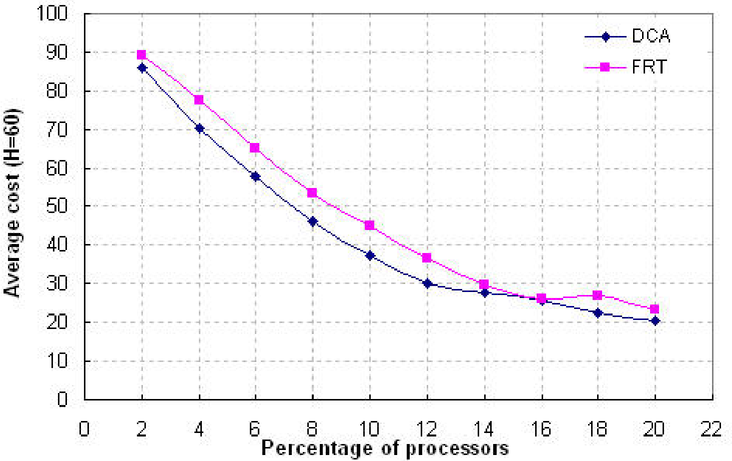
$\ell \longleftarrow \ell + 1$

Until $\|(x^{\ell+1}, z^{\ell+1}, w^{\ell+1}) - (x^\ell, z^\ell, w^\ell)\| \leq \epsilon$.

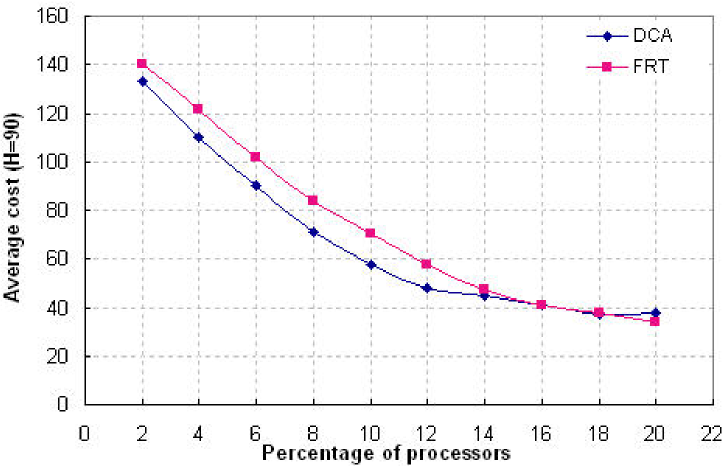
Numerical results

- We have tested our algorithms on systems with an overall number of 100 machines ($N = 100$). The parameters W_0 and W_1 are chosen as 1.0. The processor number q is varied from 2% to 20% of the machine number.
- The two horizons considered are respectively $H = 60$ days (2 months) and $H = 90$ days (3 months).
- The efficiency of DCA is compared with the algorithm based on FTR (flow time rule) algorithm
- The average cost (the mean cost) is calculated by $\frac{Totalcost}{N}$
- The average processing time (the mean time) is computed by $\frac{T_{utilization}}{q}$, where $T_{utilization}$ is the total time of processors utilization.

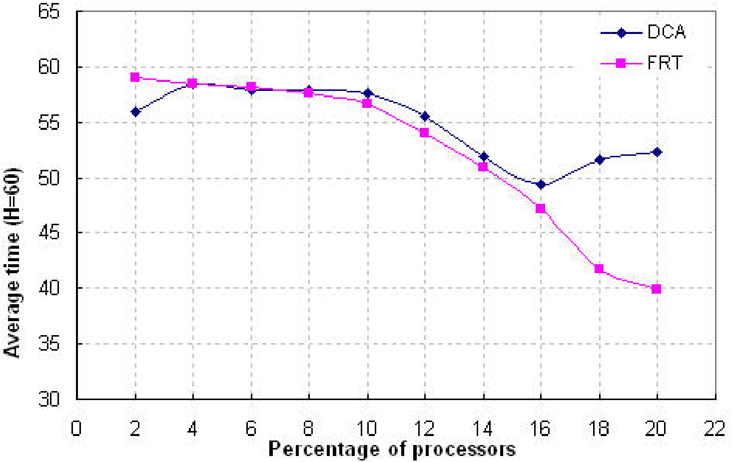
Result with H=60



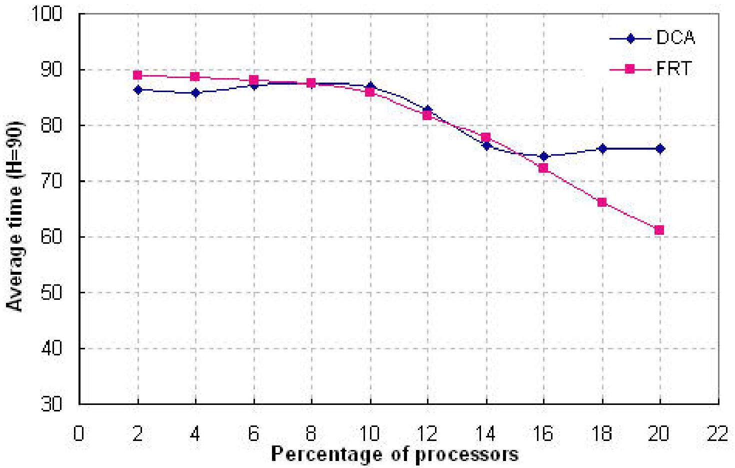
Result with H=90



Result with H=60



Result with H=90



Conclusion

- We present the discrete formulation for minimizing the maintenance cost involving flow-time and tardiness penalty. This is the firt time that a determinist model for this problem is presented.
- A new and efficient approach based on DC Programming and DCA is proposed to solve the considered problem.
- The computational results show that DCA overcomes FTR Algorithm, a recent efficient heuristic approach.
- In a future work we plan to combine DCA with global approaches.