Mixed Integer Linear Programming for Delivery Planning in Joint Replenishment Problems

Mouna Rahmouni\textsuperscript{1}\textsuperscript{2}, Jean-Claude Hennet\textsuperscript{2} and Farhat Fnaiech\textsuperscript{1} \textit{Senior member IEEE}

\textsuperscript{1}Image and Intelligent Control of Industrial Process (SICISI), ESSTT University of Tunis, Tunisia
5 Avenue Taha Hussein, 1008 Tunis
Mouna.Rahmouni@gmail.com
fnaiech@ieee.org

\textsuperscript{2}Laboratory of Sciences of Information and Systems (LSIS), Aix-Marseille University (AMU)
Avenue Escadrille Normandie Niemen - 13397 Marseille Cedex 20, France
Jean-claude.hennet@lsis.org

Abstract—This paper deals with the Joint Replenishment Problem (JRP) in multi item and multi site settings based on the common cycle strategy. The proposed delivery planning method, in deterministic and constant demand environment, uses Mixed Integer Linear Programming (MILP) algorithm. It is shown that the proposed MILP method gives better results in terms of reduced total replenishment cost than the standard delivery policy for which there is a periodicity of delivery for all the pairs (product, site).

Index Terms—Inventory theory, common cycle model, Joint Replenishment Problems.

I. INTRODUCTION

A key factor of competitiveness in markets under severe competition is the quality of the logistic links between suppliers and retailers. In particular, ordering and re-ordering policies play crucial roles in customer satisfaction and cost reduction. Considerable savings can be obtained from the reduction of inventories. However, the reordering frequency of goods is often limited by the transportation costs involved. In logistic networks, retail companies who share the same suppliers have interest to collaborate in their procurement of goods. An optimization-based answer to this issue can be derived from the resolution of the Joint Replenishment Problem (JRP).

The basic JRP model addresses the problem of how to optimally group the orders among products and/or among retailers. Typically, the JRP combines inventory problems with transportation problems. The two basic versions of the problem: one buyer – several products and one product– several buyers are naturally combined into the n products – m buyers JRP studied in this article. Demands for products are supposed deterministic and constant. Classically the delivery costs per tour include the transportation cost related to each site (major ordering cost for the site) and the additional set-up cost for each product delivered to the site (minor ordering cost). For each product and at each site, the current inventory cost is supposed to depend linearly of the quantity stored.

The classical deterministic JRP problem with one site and several products has been first formulated in the Starr and Miller book [1]. The objective is to minimize the total cost for the buyer, but some assumptions are used to define the set of eligible ordering policies. Namely, the problem consists in optimally determining a global replenishment cycle time, \(T\) and a set of integers \(k_i\) for all the items, so that the replenishment schedule of item \(i\) is supposed periodical with cycle time \(k_iT\). This model has been solved initially by Brown (see [1]), who developed a heuristic to determine the replenishment cycle time multipliers, \(k_i\) of all items \(i\).

Later on, several researchers like Goyal and co-authors [2], [3], [4], Rosenblatt and Kaspi [5] proposed other algorithms and heuristics to solve the classical JRP model. Then, the so-called “RAND algorithm” was proposed by Van Eijs [6].

For the multi item and multi buyer JRP model, Chan et al. [7] developed a genetic algorithm. Then, Li [8] proposed to extend the RAND algorithm to the multi-buyer case.

The classical policies constructed by these approaches are characterized by the value of the basic cycle time, denoted \(T\), and by the integer multipliers of \(T\) that characterize the replenishment periodicity of the various products at the different sites. However these replenishment policies are not completely determined by these parameters because of the phasing aspects of the grouping strategy. Two types of grouping strategies have been studied: the direct grouping strategy (DGS) in [3], [9], [10] and the indirect grouping strategy (IGS) [1], [2]. The approach developed in this paper is of the IGS type, with the originality of relaxing the \(k_iT\)- periodicity constraint for each item \(i\), while keeping the basic cycle time \(T\) as an elementary time period and defining a common delivery cycle time, \(pT\), with \(p\) integer valued, and using it as a delivery planning horizon.

To represent real situations in a more detailed and accurate manner, JRP models also need to integrate transport and storage capacity constraints. Several authors such as Hoque [11], Lu et al. [12] have also proposed to integrate budget constraints in the formulation of the JRP. Introduction of such restrictions complicates the problem and has justified generalizing the existing algorithms, and particularly the RAND algorithm, as in Lu et al. [12], to deal with constrained cases.
In order to solve the multi item and multi buyer JRP under transport and storage capacity constraints, this paper decomposes the JRP into three sub problems. The first sub problem consists in determining the duration $T$ of an elementary replenishment period. In the following stage, a time horizon is selected as an integer multiple, $pT$, of the elementary period. The third sub problem organizes the delivery tours for dates $T, 2T, \ldots, pT$ so as to minimize the total cost of storage, transportation and delivery over the time horizon, while satisfying demand requirements and capacity constraints. Final inventory levels are imposed equal to initial ones for each product at each site, so as to construct a replenishment policy that can be periodically repeated with periodicity $pT$.

An important concern for manufacturing networks is economic efficiency of products delivery to retailers. Hence, many parameters and variables enter into account to reduce the replenishment costs. In order to fulfill the different retailers demand and constraints, the manufacturing delivery system should implement JRP models with suitable algorithms to find a common cycle time and the optimal strategy of grouping the orders among products and/or among retailers. In order to find a common cycle time there exist many different algorithms such as genetic algorithm [7] and RAND algorithm [8], [12].

In this paper, we shall consider a given common cycle found by a RAND algorithm [12] and program the optimal grouping of deliveries with MILP over the common cycle period used as a repeated planning horizon.

This paper is organized as follows: the next section will present different JRP models that can be found in the literature and the leading algorithms for solving them. Then, in section III, the new model is presented and a decomposition technique is proposed for solving it. Section IV is dedicated to comparing the results of the new model to those previously obtained, on a numerical example of a multi buyer JRP model given in Lu et al. [12]. Finally the last section gives some conclusions and perspectives.

## II. PRESENTATION OF THE DETERMINISTIC JRP MODELS

Among the authors who analyzed JRP models, Goyal and co-authors developed an approach of the deterministic JRP model with DGS [3], [4]. Their approach consists in determining the optimal replenishment cycle per item group and the optimal grouping of items to be jointly replenished so as to minimize the total delivery cost function per time unit under perfect satisfaction of demands for all the items. In references [2], [9], [10], an IGS was proposed to solve the deterministic JRP model, through calculating the optimal basic replenishment cycle time and its multiple integer per item to minimize the total cost function.

### TABLE I

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Designation</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T$</td>
<td>basic replenishment cycle time</td>
</tr>
<tr>
<td>$k_i$</td>
<td>multiple integer per item $i$ of the replenishment cycle</td>
</tr>
<tr>
<td>$s_i$</td>
<td>minor setup or ordering cost per item</td>
</tr>
</tbody>
</table>

Van Eijs et al. [13] demonstrated that the IGS is more efficient than the DGS. This result holds true for the n products - one buyer deterministic joint replenishment problem and can be generalized to the n products - m buyers deterministic joint replenishment problem presented in Fig.1.

### Fig.1. representation of joint replenishment problems

**A. The unconstrained JRP model: n products - one buyer**

In the n products - one buyer deterministic JRP model, the supplier offers several types of products $i$ $(i=1, \ldots, n)$ that can be jointly replenished at the retailer’s. In the classical IGS setting studied by many authors ([2], [6], [9], [10], [14] to [21]), the replenishment cycle of product $i$ is characterized by its periodicity $k_i T$, which is a multiple of the replenishment cycle $T$, the integer values $k_i$ being jointly optimized for all items $i$.

Assuming a perfect satisfaction of the constant deterministic demand rates for all the items, $D_i$, the objective of the JRP is to minimize the total cost function presented in (1), with respect to decision variables $T$ and $k_i$ for $i=1, \ldots, n$.

$$TC(T,k_1,\ldots,k_n) = \frac{S}{T} + \sum_{i=1}^{n} \frac{s_i}{k_i T} + \frac{T}{2} \sum_{i=1}^{n} h_i D_i k_i$$  

(1)

Wildeman et al. [22] analyzed the JRP objective function and defined the partial average cost function per product:

$$\phi_i(k_i, T) = \frac{s_i}{k_i T} + \frac{h_i D_i k_i}{2 T}$$  

(2)

Then they showed that minimization of total cost is equivalent to the following decomposed version of the problem:

$$\inf_{T > 0} \left\{ \frac{S}{T} + \sum_{i=1}^{n} \inf_{k_i \in \mathbb{N}} \left( \phi_i(k_i, T) \right) \right\}$$  

(3)

They also provided the optimal values of multipliers $k_i$ as functions of $T$ in the form:

$$k_i(T) = \frac{\frac{1}{2} + \frac{1}{2} \sqrt{1 + \frac{8S_i}{h_i D_i T^2}}}{\frac{1}{2} - \frac{1}{2} \sqrt{1 + \frac{8S_i}{h_i D_i T^2}}}$$  

(4)
where \([\lceil \cdot \rceil]\) denotes the upper-approximate integer function.

From this result, they could reformulate the problem of total cost minimization with a single decision variable, \(T\):

\[
\text{Minimize } TC(T) = \frac{S}{T} + \frac{1}{T} \sum_{i=1}^{n} (\phi_i(k_i(T), T)) \quad (5)
\]

\(T > 0\)

A relaxed version of the problem, denoted (RJRP), could then be solved by dropping the upper-approximate integer function in (4) and allowing real positive values for multipliers \(k_i\).

B. Constrained versions of the n products - one buyer JRP

Several researchers included some constraints in the classical n products - one buyer JRP, particularly a financial resource constraint as suggested by Moon and Cha [23] and presented in problem (6):

\[
\text{Minimize } TC(T, k_1, \ldots, k_n) = \frac{S}{T} + \frac{1}{T} \sum_{i=1}^{n} \frac{s_{ij}}{k_i} + \frac{T}{2} \sum_{i=1}^{n} k_i D_i h_{ij} \quad (6)
\]

subject to: \(\sum_{i=1}^{n} D_{ki} T h_i \leq B\)

Minimum order quantity (MOQ) constraints were introduced by Porras and Dekker [24], in problem (7):

\[
\text{Minimize } TC = \frac{S}{T} + \frac{T}{2} \sum_{i=1}^{n} k_i D_i h_{ij} \quad (7)
\]

subject to: \(D_{ki} T \geq MOQ_i\).

A generic JRP model with storage capacity and investment constraints, developed by Xu et al. [25] is presented in problem (8):

\[
\text{Minimize } TC = \frac{S}{T} + \sum_{i=1}^{n} s_{ij} + \frac{T}{2} \sum_{i=1}^{n} k_i D_i h_{ij} \quad (8)
\]

subject to: \(D_{ki} T \leq g_i\)

\(D_{ki} T h_i \leq C_i\)

In the model presented in (9), Hoque [11] proposed to introduce transportation costs in the objective function of model (8).

\[
\text{Minimize } TC = \frac{S}{T} + \sum_{i=1}^{n} s_{ij} + \frac{T}{2} \sum_{i=1}^{n} k_i D_i h_{ij} + \sum_{i=1}^{n} w_i D_{ki} T \quad (9)
\]

subject to: \(D_{ki} T \leq g_i\)

\(D_{ki} T h_i \leq C_i\)

C. The n products - m buyers JRP

In the n products - m buyers deterministic JRP model, the supplier offers several types of products to several buyers (or retailers). Buyers can group their own orders for different product types with the orders from other buyers. In the classical cyclic replenishment setting, the products can be jointly replenished to several buyers or retailers and the multiples of the replenishment cycle are denoted \(k_{ij}\); each integer-valued parameter \(k_{ij}\) characterizing item \(i\) for buyer \(j\).

Thus, Chan et al. [7] and Li [8] developed such a multi-item and multi-buyer JRP model with constant demands to determine the decision variables \(T\) and \(k_{ij}\) and calculate the total cost presented in (10).

\[
TC(T, k_{ij}) = \frac{S}{T} + \frac{1}{T} \sum_{i=1}^{n} s_{ij} + \frac{T}{2} \sum_{i=1}^{n} k_{ij} D_i h_{ij} \quad (10)
\]

Anily and Haviv [26] developed a non-linear programming algorithm to solve the related cost allocation problem. Their method consists in determining first \(T\) by applying the “power of two” policy and then minimizing the JRP criterion (11).

\[
TC = \sum_{i=1}^{n} s_{ij} + \frac{T}{2} \sum_{i=1}^{n} k_{ij} D_i h_{ij} \quad (11)
\]

III. THE PROPOSED COMMON CYCLE JRP MODEL

We propose a periodic policy for the deterministic JRP multi-item and multi-site model different from the \(k_{ij} T\) policy previously described.

A. Assumptions

Before presenting the proposed JRP model, some assumptions are necessary:

1. The considered problem is multi-item and multi-site with constant deterministic demand rates for each product at each site.
2. Products are supposed independent, with no possibility of substitution.
3. The grouping strategy is indirect. It is conditioned by a limited delivery capacity at each tour and a limited storage capacity for each product at each site.
4. The elementary replenishment cycle time, \(T\), is supposed given or computed beforehand using the existing algorithms, such as RAND.
5. The common cycle \(pT\) is supposed given or computed beforehand, possibly in an iterative manner.

B. Notations

\(T\): elementary replenishment cycle time
\(h_i\): unit storage cost of product \(i\) in site \(j\)
\(x_{ijk}\): binary variable equal to 1 if demand of product \(i\) of site \(j\) is delivered in tour \(k\)
\(u_k\): binary variable equal to 1 if tour \(k\) active
\(q_{ijk}\): quantity of product \(i\) delivered to site \(j\) in tour \(k\)
\(S\): set-up cost
\(s_{ij}\): fixed cost per product \(i\) and per site \(j\)
\(z_{ijk}\): quantity of stock per product \(i\) and site \(j\) in tour \(k\)
\(Z_i\): storage capacity of product \(i\) in site \(j\) (measured in number of products)
\(v_i\): volume of product \(i\)
\(V\): delivery capacity in tour \(k\)
\(D_i\): demand rate for product \(i\) at site \(j\).

\(n\): number of products
\(m\): number of sites
\(p\): number of tours

B. Model

The objective of the addressed problem is to minimize the total replenishment cost (12) subject to demand satisfaction without any delay for all items \(i\) and sites \(j\), while satisfying constraints on delivery capacity in each tour and constraints on the storage capacity at each site.
The problem is formulated as follows:

\[
\min_{u_k, z_{ijk}} \sum_{j=1}^{m} \sum_{k=1}^{p} S_j + \sum_{i=1}^{n} \sum_{j=1}^{m} X_{ijk} S_j + \sum_{i=1}^{n} \sum_{j=1}^{m} R_k z_{ijk}
\]  \( (12) \)

Subject to:

\[\forall i, j, \sum_{k=1}^{p} q_{ijk} = D_{ij} \]  \( (13) \)

\[\forall i, j, k, \quad q_{ijk} \leq x_{ijk} D_{ij} \]  \( (14) \)

\[\forall i, j, \quad z_{ijp+1} = z_{ij1} \]  \( (15) \)

\[\forall i, j, k, \quad z_{ijk+1} - z_{ijk} - q_{ijk} = -D_{ij} \]  \( (16) \)

\[\forall i, j, k, \quad z_{ijk} \leq Z_{ij} \]  \( (17) \)

\[\forall k, \sum_{i=1}^{n} \sum_{j=1}^{m} v_{ij} q_{ijk} \leq V_k\]  \( (18) \)

\[\forall i, j, k, x_{ijk} \in \{0,1\}, u_k \in \{0,1\}, z_{ijk} \geq 0, q_{ijk} \geq 0 \]  \( (20) \)

Constraints (13) relate to each site and each product. They impose demand satisfaction over the common cycle time horizon. Constraints (14) are logical constraints that relate delivered quantities, \( q_{ijk} \), to logical variables, \( x_{ijk} \). Constraints (15) impose the p-cyclic behavior of the delivery plan. Constraints (16) are the inventory balance equations, under the assumption of demand satisfaction without any backorder, all the variables being assumed nonnegative by constraints (20) on \( z_{ijk} \) and \( q_{ijk} \), which are implicit in the Linear Programming (LP) formulation of the problem. These variables are assumed real, as it is often used in inventory planning, even if the real products are discrete units. In storage capacity constraints (17), capacity bounds are measured in numbers of products. These constraints assume a particular storing location for each product at each site. The case of storage places common to several products can also be treated through storage capacity constraints at each storage location. Constraints (18) impose a delivery capacity constraint at each tour \( k \). These resource sharing constraints create a coupling between the products to be delivered. Under such constraints, products replenishment policies cannot be independently optimized for a given value of \( T \), as in the unconstrained case. Finally, the problem formulation uses two types of logical variables, \( u_k \) and \( x_{ijk} \) and constraints (19) link these variables by indicating that condition \( u_k = 1 \) is necessary to have \( x_{ijk} = 1 \).

The proposed formulation has two major advantages. The first one is that the problem is completely linear and can be solved by any of the commercial MILP solvers. For example, the GNU Linear Programming Kit (GLPK) solver [27] is able to solve any instance of the problem if the product \( n \times m \times p \) does not exceed 25. And for larger instances, the problem run can be stopped rather rapidly, when the value of the criterion does not significantly decrease and stacks down in a constant value after successive iterations. The second advantage of this formulation is that it does not impose cyclical product deliveries within a typical common period \( \lfloor NpT/(N+1)pT \rfloor \) for any integer \( N \). Transportation capacity can then be used in the most economical manner, allowing in particular some full capacity transportation schemes impossible to achieve under cyclic policies for all products. The next section will show on a numerical example that the proposed formulation outperforms the classical policies with delivery cycles of length \( k_j T \).

Fully cyclic policies of the latter type also have the disadvantage of letting some degrees of freedom in the choice of the delivery periods, without telling the user how to use them. Typically, if \( k_j > 1 \), delivery of product \( i \) at center \( j \) is possible, for example, at dates \( 0, k_j T, 2k_j T, \ldots \) as well as at dates \( T, (k_j + 1)T, (2k_j + 1)T, \ldots \) The choice between the different possible product grouping strategies is economically very important, but out of the scope of the classical JRP, while it is central in the proposed formulation.

IV. COMPARISON BETWEEN OUR MODEL AND THE DETERMINISTIC MULTI BUYER JRP MODEL WITH ONLY BUDGET CONSTRAINT

Lu et al. [12] modified the RAND algorithm developed by Moon and Cha [23] to solve a classical version of the multi item and multi buyer deterministic JRP model, with the objective of minimizing the total cost function previously presented (10) under a financial resource constraint per cycle (21).

\[
TC = \frac{S}{T} + \sum_{i=1}^{m} \sum_{j=1}^{n} s_{ij} + \frac{T}{2} \sum_{i=1}^{m} \sum_{j=1}^{n} k_j D_{ij} h_{ij}
\]  \( (21) \)

\[\forall k, \sum_{j=1}^{n} b_j q_{ijk} \leq B\]

In this section, we compare our model results with the results obtained in Tao Lu et al. [12]. For this comparison, the model of section II is simplified by dropping the local storage capacity constraints (17) and replacing the set of delivery capacity constraints (18) by a similar set of constraints, representing a budget constraint at each delivery tour, as in (21). The common-cycle JRP model corresponding to this problem is then solved using the GLPK software [27] for Mixed Integer Linear Program (MILP). The value selected for the elementary replenishment cycle, \( T \), is the optimal value obtained in Lu et al. [12]: \( T=0.0713 \). The integer value determining the time horizon, \( p \), is selected equal to the largest value of \( k_j \) obtained in Lu et al. [12]: \( p=8 \). This parameter has been varied around this value, without any significant improvement of our results. Comparison with the results of Lu et al. [12] shows an overall cost reduction of more than 17.5%.

Table II gives the values of the input parameters. Table III provides a comparison of the results by site and
TABLE II
INPUT DATA PER ITEM AND PER SITE

<table>
<thead>
<tr>
<th>Site</th>
<th>Item</th>
<th>Demand $h_i$</th>
<th>$s_i$</th>
<th>$S$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Site 1</td>
<td>Item 1</td>
<td>10000</td>
<td>5</td>
<td>350</td>
</tr>
<tr>
<td></td>
<td>Item 2</td>
<td>5000</td>
<td>300</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Item 3</td>
<td>3000</td>
<td></td>
<td>320</td>
</tr>
<tr>
<td></td>
<td>Item 4</td>
<td>1000</td>
<td></td>
<td>400</td>
</tr>
<tr>
<td></td>
<td>Item 5</td>
<td>600</td>
<td></td>
<td>400</td>
</tr>
<tr>
<td></td>
<td>Item 6</td>
<td>200</td>
<td></td>
<td>300</td>
</tr>
<tr>
<td></td>
<td>Item 1</td>
<td>8000</td>
<td></td>
<td>250</td>
</tr>
<tr>
<td></td>
<td>Item 2</td>
<td>1000</td>
<td></td>
<td>200</td>
</tr>
<tr>
<td>Site 2</td>
<td>Item 3</td>
<td>12000</td>
<td></td>
<td>300</td>
</tr>
<tr>
<td></td>
<td>Item 4</td>
<td>6000</td>
<td></td>
<td>420</td>
</tr>
<tr>
<td></td>
<td>Item 5</td>
<td>4500</td>
<td></td>
<td>450</td>
</tr>
<tr>
<td></td>
<td>Item 6</td>
<td>100</td>
<td></td>
<td>400</td>
</tr>
</tbody>
</table>

TABLE III
COMPARISON BETWEEN THE EXPERIMENTAL RESULTS OF OUR MODEL AND OF LU ET AL. MODEL

<table>
<thead>
<tr>
<th>Sites</th>
<th>Products</th>
<th>Cost in our model</th>
<th>Cost in Lu et al. Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Site 1</td>
<td>Product 1</td>
<td>2416.74</td>
<td>2800</td>
</tr>
<tr>
<td></td>
<td>Product 2</td>
<td>1708.37</td>
<td>2400</td>
</tr>
<tr>
<td></td>
<td>Product 3</td>
<td>1493.79</td>
<td>2560</td>
</tr>
<tr>
<td></td>
<td>Product 4</td>
<td>1130.44</td>
<td>1701.67</td>
</tr>
<tr>
<td></td>
<td>Product 5</td>
<td>827.03</td>
<td>1276.26</td>
</tr>
<tr>
<td></td>
<td>Product 6</td>
<td>442.34</td>
<td>630.50</td>
</tr>
<tr>
<td>Site 2</td>
<td>Product 1</td>
<td>1929.55</td>
<td>2000</td>
</tr>
<tr>
<td></td>
<td>Product 2</td>
<td>705.02</td>
<td>1003.35</td>
</tr>
<tr>
<td></td>
<td>Product 3</td>
<td>2400</td>
<td>2400</td>
</tr>
<tr>
<td></td>
<td>Product 4</td>
<td>2480.09</td>
<td>3360</td>
</tr>
<tr>
<td></td>
<td>Product 5</td>
<td>2150.68</td>
<td>3600</td>
</tr>
<tr>
<td></td>
<td>Product 6</td>
<td>471.17</td>
<td>471.17</td>
</tr>
<tr>
<td>Cost in site 1</td>
<td>8018.71</td>
<td>11368.43</td>
<td></td>
</tr>
<tr>
<td>Cost in site 2</td>
<td>10136.51</td>
<td>12834.5</td>
<td></td>
</tr>
<tr>
<td>Total cost</td>
<td>26155.21</td>
<td>32202.9</td>
<td></td>
</tr>
</tbody>
</table>

Figures 2 to 13 compare the delivery policies obtained by the common cycle period model with the solution in [12].

Fig. 2. Delivery of the item 1 in site 1 for our model (delivery) and the algorithm by Tao Lu et al. (delivery Lu et al)

Fig. 3. Delivery of the item 2 in site 1

Fig. 4. Delivery the item 3 in site 1

Fig. 5. Delivery of the item 4 in site 1

Fig. 6. Delivery of the item 5 in site 1

Fig. 7. Delivery of the item 6 in site 1

Fig. 8. Delivery of the item 1 in site 2

Fig. 9. Delivery of the item 2 in site 2

Fig. 10. Delivery of the item 3 in site 2
V. CONCLUSION

This paper has analyzed a constrained multi-item and multi-site deterministic JRP problem. Under the assumption of constant demand rates, the natural periodicity of the solution has been translated into a common cycle approach. Deliveries have then been planned over a time horizon equal to the selected common cycle. Periodicity is then obtained by the repetition of the common cycle, but there is no need for cyclic policy within the planning horizon. Thus, the problem that has been defined is a relaxed version of the fully cyclical solution (with cycle times $k_j T$) that can be found in the literature. Accordingly, the numerical experiment has confirmed that better performance is obtained with the common cycle planning approach than with the classical fully cyclical approach.

As further work, it will be interesting to compare the proposed model with other existing models by applying advanced evolutionary programming algorithms such as genetic algorithms, neural networks etc.

REFERENCES