Instituto de Ciências Matemáticas de São Carlos

ISSN - 0103-2569

LOT-SIZING IN CAPACITATED MULTI-STAGE SERIAL SYSTEMS

VINÍCIUS AMARAL ARMENTANO
REGINA ESTHER BERRETTA
PAULO MORELATO FRANÇA

Nº 75

RELATÓRIOS TÉCNICOS DO ICMSC

São Carlos
Jun./1998

LOT-SIZING IN CAPACITATED MULTI-STAGE SERIAL SYSTEMS

Vinícius Amaral Armentano[•]* vinicius@densis.fee.unicamp.br

Regina Esther Berretta[•] regina@icmsc.sc.usp.br

Paulo Morelato França[•] franca@densis.fee.unicamp.br

[•] Faculdade de Engenharia Elétrica e de Computação
Universidade Estadual de Campinas - UNICAMP
CP 6101, 13081-970, Campinas, SP - Brazil
tel: 55 19 7883705
fax: 55 19 2891395

[•] Instituto de Ciências Matemáticas e de Computação
Universidade de São Paulo -USP
CP 668, 13560-970 , São Carlos, SP - Brazil
tel. 55 16 2739725

* To whom correspondence should be addressed

DIMENSIONAMENTO DE LOTES EM SISTEMAS MULTIESTÁGIOS SERIAIS CAPACIDADE DE PRODUÇÃO LIMITADA

RESUMO

O artigo aborda o problema de dimensionamento de lotes em sistemas multiestágios do tipo serial com limitação nos recursos para produção. Trata-se de um complexo problema de otimização que faz parte de sistemas MRP (*Material Requirements Planning*). O modelo matemático que descreve o problema utiliza o conceito de estoque de escalão e inclui *lead times*. Tempos de preparação também são considerados e isso implica que o problema de encontrar uma solução factível é NP-Completo.

É proposto um método heurístico que determina um plano de produção com objetivo de minimizar custos de estoque, produção e de preparação e que as restrições de limitação nos recursos sejam respeitadas. A heurística tem como início uma solução para o problema com capacidade infinita de produção obtida a partir da aplicação sequencial do algoritmo de Wagner-Whitin. A partir dessa solução, busca-se uma factível a partir de transferências de partes da produção entre períodos.

Testes computacionais foram feitos com 1800 exemplos numéricos considerando 40 itens e 18 períodos. Os resultados mostram soluções factíveis obtidas em 84.4% dos exemplos. Onde não foi possível obter solução factível, a heurística encontra soluções com excesso de utilização de recursos muito baixo. A qualidade das soluções é avaliada através da comparação com um limitante inferior obtido por Relaxação Lagrangiana e a média de *gap* foi menor que 10%.

Palavras-chave: dimensionamento de lotes, sistemas de produção multiestágios, sistemas seriais, heurística.

LOT-SIZING IN CAPACITATED MULTI-STAGE SERIAL SYSTEMS

ABSTRACT

The lot-sizing problem in capacitated multi-stage systems with a serial product structure is addressed. This is a complex optimization problem that is part of the decision set in Material Requirements Planning (MRP) systems. The mathematical model that describes the problem uses the concept of echelon stock and includes lead times. Setup times are taking into account which implies that the problem of finding a feasible solution is NP-Complete. This paper proposes a heuristic method that provides a production plan in order to minimize inventory, production, and setup costs. The heuristic starts from a solution for the uncapacitated problem which is given by the sequential application of the Wagner-Whitin algorithm. Feasibility is then attempted by shifting production amounts between periods. Computational tests conducted in 1800 instances with up to 40 components and 18 periods have shown that feasible solutions were obtained in 84.4% of the instances. For the infeasible instances the heuristic is able to find solutions with very low capacity excess. The solutions' quality is evaluated through a lower bound provided by Lagrangean relaxation and on average the gap is less than 10%.

Keywords: lot-sizing, multi-stage production systems, serial systems, heuristic.

1. INTRODUCTION

A methodology widely used in multi-stage production systems is Material Requirements Planning (MRP). An important task in such systems consists of the determination of the lot size of components in each period of a finite horizon so that forecast demands are met. Taking into account the production or purchasing lead time of each component and the product structure, the MRP provides a synchronized production plan of the components (Volmann et al. 1988). Most MRP systems generate production plans which ignore capacity constraints. In addition, MRP plans rarely consider setup, production and inventory costs when deciding lot sizes. As a consequence, the production plans generated are usually capacity infeasible and uneconomical.

This paper focuses on problems in multi-stage serial capacitated lot-sizing systems (MSCLS), in order to obtain low cost feasible production plans. The problem tackled here includes setup times, setup costs and component safety lead times. As pointed out by Billington et al. (1986), the notion of safety lead time is the unavoidable time from the moment a component is produced until it is available for consumption. For instance, this could be the time for paint to dry or hot metal to cool. It should be noted that during this time there is no resource consumption. When lead time is incorporated, the model and the solution methods become more complex due to the need of synchronizing the production planning periods of the components. In this case, a component can only be planned when its predecessor is available for consumption.

Florian et al. (1980) have shown that the single-item capacitated lot-sizing problem without setup time is NP-hard which implies that the MSCLS problem is also NP-hard. Furthermore, when setup times are considered the decision problem associated with the existence of a feasible solution is NP-Complete. For these reasons most methods proposed in the literature are heuristics.

Most of the research that has been conducted so far addresses the problem with zero setup times. In this class of problems few optimal algorithms have been proposed (Steinberg and Napier 1980; Chiu and Lin 1988). However, there has been a greater development of heuristic methods. For uncapacitated problems, Blackburn and Millen (1982) introduce modified costs in the objective function in order to represent the interrelation of items in assembly-type product structures and then apply mono-stage procedures to the modified problem. In another work of the same authors (Blackburn and Millen 1984), the modified cost technique is generalized to take into account capacity constraints in different stages. Using the same modification, Maes and Van Wassenhove (1991) propose heuristics for serial structures and in Billington et al. (1994) different mono-stage heuristics are compared.

Maes et al. (1991) solve the capacitated problem by linear programming (LP) and propose heuristics to round the binary variables associated to setup decisions. Kuik et al. (1993) propose similar LP-relaxation combined with simple implementations of Simulated Annealing and Tabu Search. The application of Simulated Annealing is based on the implementation of Kuik and Salomon (1990) for the uncapacitated problem.

Roll and Karni (1991) present a model for multistage lot-sizing problems with unit lead time for all components. The method starts by finding a solution by the sequential application of the uncapacitated single-item Wagner-Whitin algorithm (Wagner and Whitin 1958), and then applies a heuristic based on shifting production amounts between periods. The heuristic consists of 9 types of production shifts, four of them attempting to find a feasible solution and the rest trying to find better cost solutions.

Focusing on problems with setup times, it can be noted that there are only a few articles proposed in the literature. The paper of Billington et al. (1983) presents a mathematical model that considers setup times and proposes a compression technique for the reduction of the size of the product structure, but it does not effectively provide a solution method. In another work, Billington et al. (1986) study the multi-stage problem with constrained production capacity in only one work center (bottleneck). The model which considers setup times, setup costs and component safety lead times is solved by a heuristic based on Lagrangean relaxation. Toklu and Wilson (1992) propose a greedy heuristic for solving special cases of the model presented by Billington et al. (1986). In a recent article, Harrison and Lewis (1996) present a heuristic for serial assembly systems with multiple constrained resources without taking into account setup costs. The method fixes the binary variables associated with the setup times and then solves a sequence of linear programming models.

Clark and Armentano (1993) present a mathematical model for the general structure case based on that proposed by Billington et al. (1983), using the concept of echelon stock. In another article, Clark and Armentano (1995) present a heuristic method to solve the problem as formulated in Clark and Armentano (1993) which starts by finding a solution generated by the sequential application of the Wagner-Whitin algorithm to the uncapacitated problem, and then it attempts to attain feasibility through production shifts between periods. Franca et al. (1997) improve the method presented in Clark and Armentano (1995) by introducing new iterative procedures which increase the likelihood of attaining feasibility. In contrast to the heuristic proposed by Clark and Armentano, which stops when a feasible solution is found, the improved version in able to provide several quality solutions.

This paper presents a heuristic method to provide a production plan in multi-stage serial systems considering inventory, production, and setup costs, as well as a finite amount of resources for the production and setup of each item. In addition, non-zero lead times are assumed. The heuristic starts from a solution to the corresponding uncapacitated problem and tries to obtain a feasible solution by moving production backwards and forwards in time. When these movements are not allowed, a new shift procedure which explores the dependency among components in the product structure is activated. This leads to significant improvements in the results, especially with regards to finding feasible solutions. The paper is organized as follows. In section 2, the problem is modeled through the use of echelon stock.

Section 3 describes the proposed heuristic method as well as an illustrative example. Finally, computational results and conclusions are presented in section 4 and 5, respectively.

2. MATHEMATICAL FORMULATION

In this section, the MSCLS problem is formulated as a mixed-integer programming model using the concept of echelon stock (Clark and Scarf 1960; Afentakis et al. 1984) in a rolling horizon scheme. The components are numbered from 1 to N and the final item is assumed to be the number 1. Consider the following mathematical notation:

Variables:

x_{it} lot-size of component i in period t.

y_{it} 1, if component i is produced in period t; 0, otherwise.

I_{it} inventory stock of component i at the end of period t

E_{it} level of echelon stock of component i at the end of period t.

Parameters:

N number of components.

T number of planning periods.

K number of resources.

c_{it} unit production cost of component i in period t.

s_{it} setup cost incurred if component i is produced in period t.

e_{it} unit cost of echelon stock of component i at the end of period t.

v_{ikt} unit amount of resource k used in the production of component i in period t.

f_{ikt} fixed amount of resource k used in the setup of component i in period t.

b_{kt} amount of resource k available in period t.

U_{it} upper bound for x_{it}.

D_{it} echelon demand of component i in period t.

r_{ij} number of components i needed by one unit of the successor component j.

L(i) lead time of component i, ensuring that the lot x_{it} is available for consumption only at the beginning of period t+L(i).

T(i)+1 the period at which, on a rolling horizon of T periods, production planning of component i must start in order to satisfy demand for its immediate successor at periods $t \geq T(i)+1+L(i)$ and eventually with that of the end-item. The lot x_{it} belongs to one of the following three categories: (1) if $t \leq T(i)$ then x_{it} is known from previous rolling horizon applications of the model and has a fixed value; (2) if $T(i)+1 \leq t \leq T(i)+T$ then x_{it} is a model decision variable; (3) if $t > T(i)+T$ then x_{it} is beyond the planning horizon and will enter the model only in future rolling horizon application. Further details about T(i) and how it is determined can be found in Clark and Armentano (1993).

Echelon stock is defined as:

$$E_{it} = \begin{cases} I_{i,L_i+t} + r_{i,i-1} E_{i-1,L(i)+t} & i \neq 1 \\ I_{i,L(i)+t} & i = 1 \end{cases}$$

In words, E_{it} is the total system stock of component i at the end of period t, both as a stand-alone component, $I_{i,L(i)+t}$, and as part of the successor component $r_{i,i-1}E_{i-1,L(i)+t}$.

Similarly, echelon demand is defined as the total demand of an component, i.e., its own independent demand plus the echelon demand of the successor component:

$$D_{it} = \begin{cases} d_{i,L_i+t} + r_{i,i-1} D_{i-1,L(i)+t} & i \neq 1 \\ d_{i,L(i)+t} & i = 1 \end{cases}$$

The mathematical model below is a specialization for the serial structure case of the model proposed in Clark and Armentano (1993).

$$(ME): \text{minimize} \sum_{i=1}^{N} \sum_{t=T(i)+1}^{T(i)+T} [s_{it}\, y_{it} + c_{it}\, x_{it} + e_{it}\, E_{it}] + \sum_{i=1}^{N} \sum_{t=1}^{T(i)} e_{it}\, E_{it} \tag{1}$$

subject to

$$E_{i,t-1} + x_{it} - E_{it} = D_{it} \qquad i = 1, ..., N \; ; \; t = 1+T(i), ..., T+T(i) \tag{2}$$

$$r_{i,i-1}\, E_{i-1,t+L_i} \leq E_{it} \qquad i = 1, ..., N \; ; \; t = 1+T(i), ..., T+T(i) \tag{3}$$

$$\sum_{i/t \leq T(i)+T} [f_{ikt}\, y_{it} + v_{ikt}\, x_{it}] \leq b_{kt} \qquad k = 1, ..., K \; ; \; t = 1, ..., T+T(1) \tag{4}$$

$$x_{it} \leq U_{it} y_{it} \qquad i = 1, ..., N \; ; \; t = 1+T(i), ..., T+T(i) \tag{5}$$

$$x_{it} \geq 0, E_{it} \geq 0 \text{ e } y \in \{0,1\} \qquad i = 1, ..., N \; ; \; t = 1+T(i), ..., T+T(i) \tag{6}$$

The objective function consists of the minimization of setup, production, and inventory costs involving only the decision periods of each component. Due to the fact that the model is applied on a rolling horizon basis, the variables are those between T(i)+1 to T(i)+T. The values of x_{it}, y_{it}, and E_{it} for $t \leq T(i)$ are fixed, as they were decided in the previous application of the model. It is shown (Clark and Armentano 1993) that the constant term $\sum_{i=1}^{N-1} \sum_{t=1}^{T(i)} e_{it}\, E_{it}$ is due to the equivalence between models using conventional and echelon stocks. Constraints (2) represent the inventory balance equations. The inter-echelon constraints (3) follow from the definition of echelon stock and the non-negativity of the conventional stock I_{it}. The capacity constraints in (4) state that the fixed and variable resources used for production are limited to the resource amount available in period t. Constraints (5) provide an upper bound for x_{it}, if $y_{it} = 1$, and ensure that $x_{it}=0$, if $y_{it}=0$. It can be easily shown that $E_{it}\geq 0$ in (6) is redundant and follows from (3).

3. THE HEURISTIC METHOD

The method consists of three procedures. Initially, procedure P1 assumes that the constraint set (4) is relaxed in model ME, and finds an uncapacitated solution by applying the optimal Wagner-Whitin algorithm to each component. Starting from the final-item, the algorithm is then applied to components 2, 3..., N in order to obtain a production plan. As the solution found in P1 is likely to be infeasible, i.e., there may be periods when the resource amount spent is greater than the capacity available in the period, the smoothing procedure P2 tries to find a feasible solution by shifting production amounts between time periods. When no amount of any component can be transferred from a given period t, a new shifting routine tries to move quantities of other components in order to create a shift for a component in period t. The improvement procedure P3 starts from the solution found in P2 and searches for a lower cost feasible solution. In the following, the procedures P2 and P3 are described in detail.

Smoothing Procedure P2

The percentage capacity exceeded in period t can be expressed by

$$\text{Excess}(t) = 100 \sum_{k=1}^{K} \left(\sum_{i/t \leq T(i)+T} [f_{ikt}\, y_{it} + v_{ikt}\, x_{it}] - b_{kt} \right)^{+} \Big/ b_{kt} \tag{7}$$

where $a^+ = \max\{0, a\}$. Procedure P2 tries to move portions of the production from periods with Excess(t)>0 to other periods, aiming to reduce the use of resource in period t, and searching for a feasible solution. A move can be represented by (q,i,t,t'), which means that a quantity q of component i is moved from period t to period t'.

Procedure P2 is divided into a backward step in time, where periods $t=T(1)+T,\ldots,2$ are analyzed in that order with $t' < t$, and a forward step where periods are analyzed in the order $t = 1,2,\ldots,T(1)+T-1$ with $t' > t$. In each of these steps and at each period where $\text{Excess}(t) > 0$, the procedure analyzes shifts (q,i,t,$t\ell$) for those components i which are produced in period t. The periods $t\ell$ are those between t and τ, where

$$\tau = \begin{cases} \max\{1, \text{ last period in which there is production of component } i\} & \text{if } t\ell < t \\ \min\{T(1)+T, \text{ first period in which there is production of component } i\} & \text{if } t\ell > t \end{cases}$$

The amount of the component to be shifted is determined in such a way as to satisfy constraints (2) and (3). For each component i that is produced in an infeasible period t, first determine $M_{i,t,t\ell}$, which represents the maximum amount of component i that can be moved from period t to $t\ell$ in such a way that the inter-echelon feasibility is preserved. It is also checked whether there is an amount Q_{itk} smaller than $M_{i,t,t\ell}$ that reduces Excess(t) to zero for a given resource. The choice between these two quantities is based on a cost index R(q,i,t,$t\ell$) calculated as a function of the total cost variation caused by the shift per unit of overload eliminated.

For a given infeasible period t, it may occur that no shifts are allowed because $M_{i,t,t\ell} = 0$, for any i and $t\ell$. In this case, the procedure calls the routine **create_shift**, which tries to perform specific shifts in periods other than t. The calculation of $M_{i,t,t\ell}$, Q_{itk}, R(q,i,t,$t\ell$) and the routine **create_shift** are detailed below:

- **$M_{i,t,t\ell}$ for $t\ell < t$**

From equation (2), when a shift (q,i,t,$t\ell$) with $t\ell < t$ is performed the inventory level of component i in periods $t\ell$ to t-1 will be added by q. In order to ensure that constraints (3) are satisfied, the quantity q must be such that

$$r_{i+1,i}\left(E_{i,\delta+L(i+1)} + q\right) \le E_{i+1,\delta} \qquad \delta = t\ell - L(i+1),\ldots,t-L(i+1)-1$$

or

$$q \le \frac{E_{i+1,\delta-L(i+1)} - r_{i+1,i}E_{i,\delta}}{r_{i+1,i}} \qquad \delta = t\ell,\ldots,t-1$$

Thus,

$$M_{i,t,tl} = \begin{cases} \min\limits_{\delta=tl,\ldots,t-1} \left\{ \dfrac{E_{i+1,\delta-L(i+1)} - r_{i+1,i} E_{i,\delta}}{r_{i+1,i}}, x_{it} \right\} & \text{if } i \neq N \\ x_{it} & \text{if } i = N \end{cases} \qquad (8)$$

- **$M_{i,t,tl}$ for $tl > t$**

Similarly, from equation (2), a shift (q,i,t,tl) reduces by q the inventory of component i in periods t to tl-1. From (3), we can write

$$r_{i,i-1}\, E_{i-1,\delta+L(i)} \leq E_{i,\delta} - q \qquad \delta = t,\ldots,tl-1$$

Thus,

$$M_{i,t,tl} = \begin{cases} \min\limits_{\delta=t,\ldots,tl-1} \{E_{i,\delta} - r_{i,i-1} E_{i-1,\delta+L(i)}, x_{it}\} & \text{se } i \neq 1 \\ \min\limits_{\delta=t,\ldots,tl-1} \{E_{i,\delta}, x_{it}\} & \text{se } i = 1 \end{cases} \qquad (9)$$

- **$Q_{i,t,k}$**

The quantity that reduces the overload of resource k in a given period t can be expressed by

$$Q_{i,t,k} = \left(\sum_{j=1}^{N} \left(f_{jkt}\, y_{jt} + v_{jkt}\, x_{jt} \right) - b_{kt} \right)^{+} / v_{ikt}, \text{ if } Q_{i,t,k} < M_{i,t,tl}$$

- **R(q,i,t, tl)**

As mentioned earlier, the choice of the amount to be shifted is based on the calculation of the cost index R(q,i,t, tl) for each one of the candidate shifts (q,i,t, tl).

$$R(q,i,t,tl) = \frac{\Delta cost + \beta \,.\, Penalty}{Excess\_decrease} \qquad (10)$$

Δcost is the ratio Additional_cost/Total_cost, where Additional_cost is the cost change due to the shift (q,i,t,tl) and Total_cost corresponds to the current production and inventory cost for all components. The expression for Additional_cost is as follows

$$\text{Additional\_cost} = su_1 - su_2 + q\left[(c_{it\lambda} - c_{it}) + \begin{cases} \sum_{\delta=tl,\dots,t-1} e_{i\delta} & \text{if} \quad tl < t \\ \sum_{\delta=t,\dots,tl-1} -e_{i\delta} & \text{if} \quad tl > t \end{cases}\right]$$

where,

$$su_1 = \begin{cases} s_{it\lambda} & \text{if } x_{it\lambda} = 0 \\ 0 & \text{otherwise} \end{cases} \qquad su_2 = \begin{cases} s_{it} & \text{if } q = x_{it} \\ 0 & \text{otherwise} \end{cases}$$

The Penalty term is a non-negative quantity which can be interpreted as a cost for overuse of resources in periods t and tl. and β is a control parameter. It is defined as

$$\text{Penalty} = \text{Excess\_after}(t) + [\ \text{Excess\_after}(tl) - \text{Excess\_before}(tl)\ ]$$

where,

$$\text{Excess\_before}(t) = \text{Excess}(t) \text{ before the move.}$$

$$\text{Excess\_after}(t) = \text{Excess}(t) \text{ after the move.}$$

The cost index denominator Excess_decrease is given by

$$\text{Excess\_decrease} = \text{Excess\_before}(t) - \text{Excess\_after}(t)$$

We call cycle a sequence of a backward and a forward step. During the first cycle β =1, and if a feasible solution is not found, β is increased by 1 in the subsequent cycles. In this way, the increase of parameter β in the smoothing procedure gives greater importance to the overuse of resources. If no feasible solution is found after a prespecified number of cycles, the procedure fails.

The shift (q,i,t,tl) that yields the smallest value of R(q,i,t,tl) is performed and new shifts are executed while Excess(t) is greater than zero.

- **Create_shift**

When $M_{i,t,tl} = 0$ for all items i and all periods tl, no shift is allowed from a given period t. Thus, the procedure **create_shift** tries to alter the echelon stock of other components in specific periods so that $M_{i,t,tl} > 0$.

Similarly to the calculation of $M_{i,t,tl}$, the routines **create_shift** for tl < t and tl > t are distinct.

a) Create_shift_backward ($tl < t$)

Consider the shift (q,i,t,tl), where tl = t-1, $x_{it} > 0$, and $i \neq N$. If $M_{i,t,t-1} = 0$, from (8) it follows that

$$E_{j,t-1-L(j)} - r_{j,i} E_{i,t-1} = 0 \qquad j = i+1$$

In order to have $M_{i,t,t-1} > 0$, the echelon stock $E_{j,t-1-L(j)}$ should be increased. For this, a shift from a period greater than t-1-L(j) to the period t-1-L(j) is required. A shift (q,j,t-L(j),t-1-L(j)) is attempted in order to simplify the routine. If such a shift exists, it is performed and therefore $E_{j,t-1-L(j)}$ is increased and $M_{i,t,t-1}$ becomes greater than zero. If this shift is not possible because $M_{j,\ t-L(j),\ t-1-L(j)} = 0$, the same procedure is repeated, and a shift for component j is attempted by examining its predecessor component. As shown below, the routine is recursive, and it succeeds if $M_{i,t,t-1} > 0$; otherwise it fails.

```
Create_shift_backward ( i, t )
{
        if i=N then return(failure)
        else {
                j = i+1
                tj = t -L(j)
                if T(j)+1 < tj ≤ T(j)+T and xj,tj>0 then {
                        if Mj,tj,tj-1 = 0         then    result = Create_shift_backward ( j, tj )
                                                  else    result = success
                        if result = success
                                do the shift (Mj,tj,tj-1, j, tj, tj-1)
                                return(success)
                else return(failure)
                }
        else return (failure)
        }
}
```

b) Create_shift_forward (tl > t)

Analogously, consider the shift (q,i,t,tl), where $tl = t+1$, $x_{it} > 0$, $E_{i,t} > 0$ and $i \neq 1$. If $M_{i,t,t+1} = 0$, then from (9)

$$E_{i,t} - r_{ij} E_{j,t+L(i)} = 0 \qquad j = i-1$$

In the forward routine, the echelon stock $E_{j,t+L(i)}$. should be decreased in order to have $M_{i,t,t-1} > 0$. Hence, it is necessary to move component j from a period t+L(i) to a later period so as to reduce the echelon stock $E_{j,t+L(i)}$. A shift (q, j, t+L(i), t + L(i)+1) is attempted to simplify the procedure. If such a shift exists, it is performed and therefore $E_{j,t+L(i)}$ is reduced and $M_{i,t,t+1}$ becomes greater than zero. If this shift is not possible due to $M_{j,t+L(i),t+L(i+1)} = 0$, a shift for component j is attempted by examining its successor component. The pseudo-code follows.

```
Create_shift_forward ( i, t )
{
        if i=1 then return (failure)
        else {
                j = i-1
                tj = t+L(i)
                if T(j)+1 ≤ tj < T(j)+T and xj,tj > 0 then {
                        if Mj,tj,tj+1 =0        then    result = Create_shift_forward ( j , tj )
                                                else    result = success
                        if result = success
                                do the shift (Mj,tj,tj+1, j, tj, tj+1)
                                return(success)
                        else return(failure)
                        }
                else return (failure)
                }
}
```

Note that the routine **create_shift** fails if: (1) item j is not within its planning periods; (2) there is no production of item j in period tj; (3) the attempt to move item j is unsuccessful.

The backward and forward steps are performed consecutively until a feasible solution has been found or a stopping criterion has been reached. In each of the steps, routine **create_shift** is run every time that it is not possible to perform shifts due to $M_{i,t,tl}$ being zero.

Improvement Procedure P3

If procedure P2 yields a feasible solution, then procedure P3 is applied in an attempt to find a lower cost solution. This procedure is similar to P2, also employing the backward and forward steps. However, during procedure P3 we only allow feasible and improving solutions. The maximum quantity to maintain capacity feasibility for a shift (q,i,t,tʹ) can be calculated as

$$F_{i,t,\tau} = \min\left\{M_{i,t,\tau}; \min_{k=1,\dots,K}\left(G_{i,k,\tau}\right)\right\}$$

where

$G_{i,k,\tau}$ = maximum quantity of item i that can be moved from period t to period τ so as not to violated the capacity constraint in τ. For a given resource k, such a quantity is given by

$$G_{i,k,\tau} = \begin{cases} \dfrac{J_{k\tau} - f_{ik\tau}}{v_{ik\tau}} & \text{if} \quad x_{it} = 0 \\ \dfrac{J_{k\tau}}{v_{ik\tau}} & \text{if} \quad x_{it} > 0 \end{cases}$$

where

$$J_{k,\tau} = b_{k,\tau} - \sum_{i=1}^{N}\left(f_{ik\tau}y_{i\tau} + v_{ik\tau}x_{i\tau}\right)$$

represents the capacity slack of resource k in period τ.

Two quantities are analyzed in a shift (q,i,t,tʹ): q = $F_{i,t,t\prime}$ and q sampled from the uniform distribution U[1, $F_{i,t,t\prime}$]. Computational tests have shown that the inclusion of this random quantity leads to better solutions. Among all candidate shifts (q,i,tʹ) in a given period t we choose the one which minimizes the ratio Δcost as defined in (10). As the procedure searches for improving solutions, only negative values of Δcost are accepted. The backward and forward steps are performed consecutively until no improving solution is found.

The heuristic proposed in this article, denoted FAB, uses a similar approach to that suggested in Clark and Armentano (1995) and in Franca et al (1997). However, the proposed method introduces new features which proved to be quite effective when dealing with serial systems. In Clark and Armentano (1995) the procedure P2 consists of a single pass backward step and the choice of the best move does not make use of a penalty term. The inclusion of the **creat_shift** routine made it possible to increase significantly the number of feasible solutions as shown in section 5. Finally, our method generalizes the improvement procedure P3 proposed in Franca et al. (1997) to problem with nonzero lead times

4. AN EXAMPLE

Consider a serial structure with 3 components (N=3) constrained by one resource (K=1) over a 4-period planning horizon (T=4) with parameters shown in Tables 1 and 2.

Table 1. Parameters for the example

L(i)	1
c_{it}	1
s_{it}	100
e_{it}	2
f_{it}	150
v_{it}	1
d_{it}	50 for i = 1 5 for i = 2 and 3

Table 2.Resource availability

t	1	2	3	4	5	6
b_t	500	500	600	600	400	300

From the component lead times the T(i) values are given by T(1)=2, T(2)=1 and T(3)=0. These values show that the planning of component 3 begins in period 1 (T(3)+1) and is available to be used in period 2. Therefore, component 2 can be planned in periods 2, 3, 4 and 5 and component 1 in periods 3, 4, 5, and 6.

Table 3 shows the production quantities obtained by the sequential application of Wagner-Whitin algorithm and the corresponding echelon stock and echelon demand.

Table 3. Initial solution

Period	1	2	3	4	5	6
D_{1t}	50	50	50	50	50	50
x_{1t}	50	50	50	50	50	50
E_{1t}	0	0	0	0	0	0
D_{2t}	55	55	55	55	55	*
x_{2t}	0	55	55	55	55	*
E_{2t}	55	0	0	0	0	*
D_{3t}	60	60	60	60	*	*
x_{3t}	60	60	60	60	*	*
E_{3t}	0	0	0	0	*	*

Table 5. Feasible solution at the end of the smoothing procedure

Period	1	2	3	4	5	6
D_{1t}	50	50	50	50	50	50
x_{1t}	50	50	50	50	100	0
E_{1t}	0	0	0	0	50	0
D_{2t}	55	55	55	55	55	*
x_{2t}	0	110	0	110	0	*
E_{2t}	55	0	55	0	55	*
D_{3t}	60	60	60	60	*	*
x_{3t}	120	0	120	0	*	*
E_{3t}	60	0	60	0	*	*

Assume that the values in the shading cells have been obtained by previous applications of the model. The asterisks in Table 3 are associated to values which will be obtained in future applications of the model.

It is easy to see that the initial solution is capacity infeasible. Table 4 shows the application of the smoothing procedure and the resulting feasible solution is presented in Table 5.

Table 4. Smoothing procedure for the example

Backward Step		
t = 6	Excess(6) = ((50*1 + 150)-300) $^+$/300 = 0	
t = 5	Excess(5) = ((50+55)*1 + 2*150 - 400) $^+$ / 400 = 1.3%	
	i = 1 $\quad$ tt = 4 $\quad$ $M_{1,5,4}$ = 0	
	i = 2 $\quad$ tt = 4 $\quad$ $M_{2,5,4}$ = 0	
	every $M_{i,5,tt}$ = 0 ⇒ Create_shift_backward (i=2 , t=5)	
	j=3, tj=4, $M_{3,4,3}$= 60 ⇒ success	**move q=60 of i=3 from t=4 to t=3.**
	$M_{2,5,4}$ = 55 $\quad$ $R_{(55,2,5,4)}$=0.43	**move q=55 of i=2 from t=5 to t=4.**
t = 4	Excess(4) = (600 - ((50+110)*1+300)) $^+$ = 0	
t = 3	Excess(3) = ((50+55+120)*1+450-600) $^+$ /600 = 12.5%	
	i = 1 $\quad$ tt is not a planning period; no shift is allowed	
	i = 2 $\quad$ tt = 2 $\quad$ $M_{2,3,2}$ = 0	
	i = 3 $\quad$ tt = 2 $\quad$ $M_{3,3,2}$ = 120 $\quad$ $R_{(120,3,3,2)}$=2.51	
	$Q_{3,3,1}$ = 75 $\quad$ $R_{(75,3,3,2)}$ =1.83	**move q=75 of i=3 from t=3 to t=2**
t = 2	Excess(2) = ((50+55+135)*1+450-500)$^+$/500 =38%	
	i = 1 $\quad$ does not belong to this planning period	
	i = 2 $\quad$ tt is not a planning period; no shift is allowed	
	i = 3 $\quad$ tt = 1 $\quad$ $M_{3,2,1}$ = 135 $\quad$ $R_{(135,3,2,1)}$=0.46	**move q=135 of i=3 from t=2 to t=1**
Forward Step		
t = 1	Excess(1) = ((50+195)*1+300 - 500)$^+$/500 = 9%	
	i = 3 $\quad$ tt = 2 $\quad$ $M_{3,1,2}$ = 135 $\quad$ $R_{(135,3,1,2)}$ = 3.37	
	$Q_{3,1,1}$ = 45 $\quad$ $R_{(45,3,1,2)}$ = 2.27	
	tt = 3 $\quad$ $M_{3,1,3}$ = 75 $\quad$ $R_{(75,3,1,3)}$ = -0.12	**move q=75 of i=3 from t=1 to t=3**
	$Q_{3,1,1}$ = 45 $\quad$ $R_{(45,3,1,3)}$ = -0,07	
t = 2	Excess(2) = (500 - ((50+55)*1+300))$^+$ = 0	
t = 3	Excess(3) = ((50+55+120)*1+450 - 600) $^+$/600 = 12.5%	
	i = 1 $\quad$ tt = 4 $\quad$ $M_{1,3,4}$ = 0	
	i = 2 $\quad$ tt = 4 $\quad$ $M_{2,3,4}$ = 0	
	i = 3 $\quad$ tt = 4 $\quad$ $M_{3,3,4}$ = 5 $\quad$ $R_{(5,3,3,4)}$ = 22.7	**move q=5 of i=3 from t=3 to t=4**
	Excess(3) > 0	
	i = 1 $\quad$ tt = 4 $\quad$ $M_{1,3,4}$ = 0	
	i = 2 $\quad$ tt = 4 $\quad$ $M_{2,3,4}$ = 0	
	i = 3 $\quad$ tt = 4 $\quad$ $M_{3,3,4}$ = 0	
	$M_{i,3,tt}$ = 0 ⇒ $\quad$ Create_shift_forward (i=2,t=3)	
	j=1, tj=4, $M_{1,4,5}$ = 0 ⇒ Create_shift_forward (i=1,t=4) ⇒ failure (i=1)	
	Create_shift_forward (i=3,t=3)	
	j=2, tj=4, $M_{2,4,5}$ = 55 ⇒ success	**move q=55 of i=2 from t=4 to t=5**
	$M_{3,3,4}$ = 55 $\quad$ $R_{(55,3,3,4)}$ = -0,06	**move q=55 of i=3 from t=3 to t=4**
	Excess(3) > 0	
	$M_{i,3,tt}$ = 0 ⇒ $\quad$ Create_shift_forward (i=2,t=3) ⇒ failure	
	Create_shift_forward (i=3,t=3) ⇒ failure	
	It is not possible to eliminate the excess	
t = 4	Excess(4) = 2.5 % $\quad$ No shifts were possible in this period	
t = 5	Excess(5) = 1.2 % $\quad$ No shifts were possible in this period	
Backward Step		
t = 6	Excess(6) = 0	
t = 5	Excess(5) = 1.2%	
	$M_{i,5,tt}$ = 0 ⇒ $\quad$ Create_shift_forward (i=2, t=5)	
	j = 3, tj = 4, $M_{3,4,3}$ = 60 ⇒ success	**move q=60 of i=3 from t=4 to t=3**
	$M_{2,5,4}$ = 55 $\quad$ $R_{(55,2,5,4)}$ = 0.42	**move q=60 of i=3 from t=4 to t=3**
	Excess(5) = 0	
t = 4	Excess(4) = 0	
t = 3	Excess(3) = 12.5%	**move q=55 of i=2 from t=3 to t=2**
t = 2	Excess(2) = 0	

5. COMPUTATIONAL RESULTS

The heuristic was coded in C language and tests were conducted on a SUN SPARCstation 20. Instances were generated according to the dimensions shown in Table 6, totaling 30 different sizes. Data for these instances were generated from a uniform distribution in the intervals presented in Table 7. Parameter r_{ij} is a constant fixed at value 1 and L(i) was randomly generated from the values 0 or 1.

Table 6. Dimensions of the instances

N	3, 6, 10, 20,40
T	6, 12,18
K	1, 2

Table 7. Parameters generated from a uniform distribution U[a,b]

Parameter	Interval [a,b]	Observation
c_{it}	[1.5,2]	
s_{it}	[5,95] [50,950]	low setup cost high setup cost
e_{it}	[0.2,4]	
v_{ikt}	[150,200] [200,300]	k = 1 k = 2
f_{ikt}	[1.5,2] [2,3]	k = 1 k = 2
d_{it}	[0,18] [0,180]	$i \neq 1$ i = 1

The available capacity for each resource k in each period t (b_{kt}) is calculated from the solution obtained with the application of a lot-for-lot policy which consists of producing in each period t the demand of this period. Initially, the total amount of each resource used in this solution is calculated as

$$B_k = \sum_{i=1}^{N} \sum_{\substack{t=1 \\ D_{it}>0}}^{T(i)+T} (f_{ikt} + v_{ikt} D_{it}) \qquad k = 1,\ldots,K$$

Then the amount of resource k available in each period is

$$B_{kt} = \frac{B_k}{T(1) + T} \; p_t$$

where p_t is defined as

$$p_t = \frac{\text{number of components i in period t / t} \leq \text{T(i) + T}}{N}$$

The capacity b_{kt} is defined as the ratio B_{kt}/α where α assumes the values 0.9, 1.0, and 1.1, representing loose, normal, and tight capacity levels, respectively.

In order to exemplify the capacity calculations, consider a solution obtained by the application of the lot-by-lot policy to the example introduced in section 4. Using the data in Table 3, the amount B_1 = 3065 and B_{1t} for t=1,...,6 is given by 3065/6 , 3065/6 , 3065/6 , 3065/6 , (3065/6) 2/3 , (3065/6) 1/3), that is, (511, 511, 511, 511, 341, 170). Note that, p_5 = 2/3 and p_6 = 1/3 since there are two components planned in period 5 and one component in period 6.

For each size (Table 6), each setup cost (Table 7) and each capacity level, 10 different seeds were used to generate a total of 1800 instances. The heuristic's performance is analyzed with respect to 2 factors, setup cost and capacity level, resulting in 6 instance types, as shown in Table 8.

Table 8. Type description

Type	Setup cost	Capacity level
1	low	1.1
2	high	1.1
3	low	1.0
4	high	1.0
5	low	0.9
6	high	0.9

Results are grouped according to instance types (Figures 1 and 2) and instance sizes (Figures 3, 4 and 5). Figure 1 shows the percentage of feasible solutions (FEA) that the heuristic was able to find with respect to the total number of instances. The notation used follows:

CA heuristic by Clark and Armentano

FAB heuristic proposed in this article

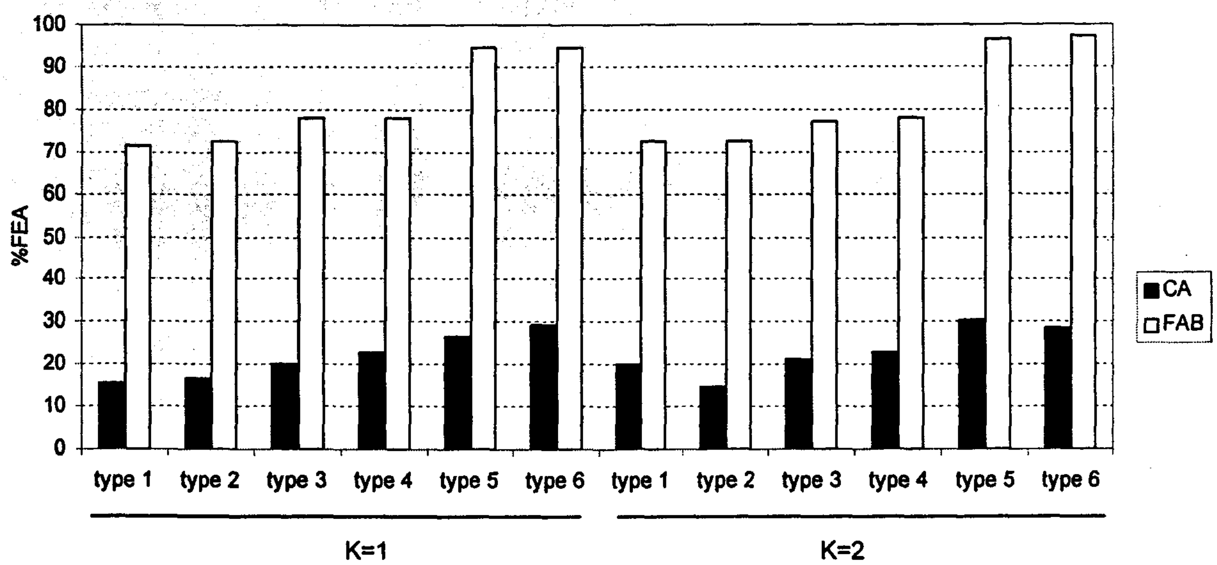

Figure 1. Percentage of feasible solutions found by the heuristic

As expected, the heuristic is able to find a higher percentage of feasible solutions for instances with loose capacity (types 5 and 6). For such types the heuristic attained a feasibility degree of 90% as compared to a degree of 30% of the heuristic CA.

Figure 3 presents the average percentage of overuse of the resources (OVER) of the initial solution given by the application of the Wagner-Whitin algorithm and the final solutions obtained by the heuristics CA and H. Such a percentage can be viewed as a measure of the infeasibility degree of a solution and is expressed by

$$OVER = 100\left(\sum_{t=1}^{T(1)+T} Excess(t) / T(1) + T\right)$$

where Excess(t) is given by (7).

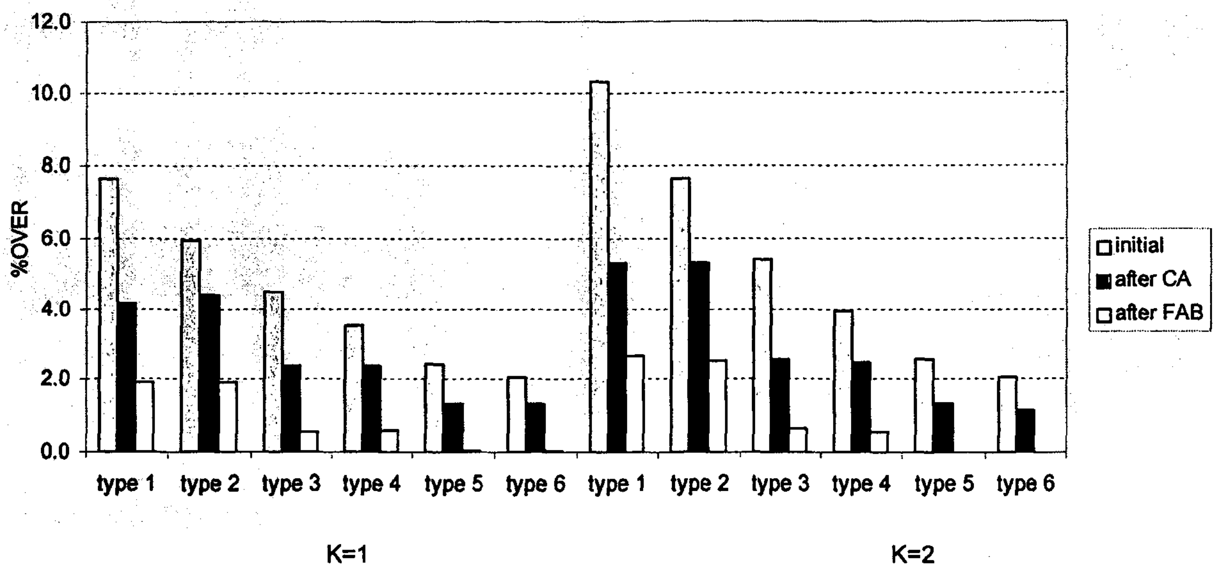

Figure 2. Infeasibility degree of the initial and heuristic solutions

Note that for instances with tight capacity (types 1 and 2) the maximum average infeasibility degree is less than 3% and for the other capacity levels this degree decreases to less than 1%. As shown in Figures 1 and 2, the performance of the heuristic concerning the indexes FEA and OVER is almost not affected by the number of resources.

Figures 3 and 4 show that for instances with up to 20 items, the heuristic is able to find feasible solutions for almost all instances, and the degree of infeasibility is negligible. For instances with 40 items the original capacity ratios B_{kt}/b_{kt} revealed to be too tight and for this reason were reduced to 0.7, 0.8 and 1.0. The number of infeasible solutions in this case has increased, however the associated infeasibility degree does not exceed 4.5% on average.

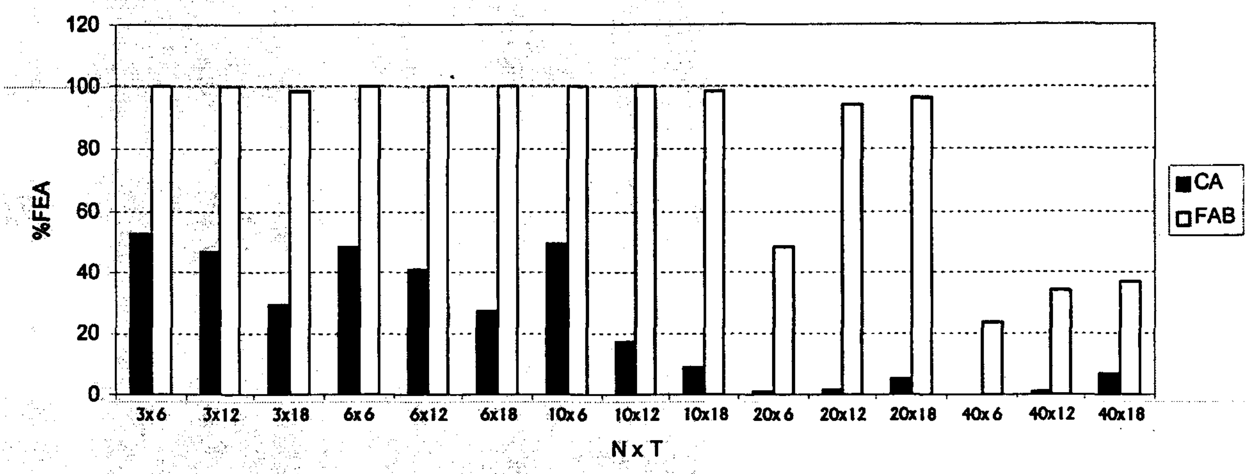

Figure 3. Percentage of feasible solutions *versus* instance size.

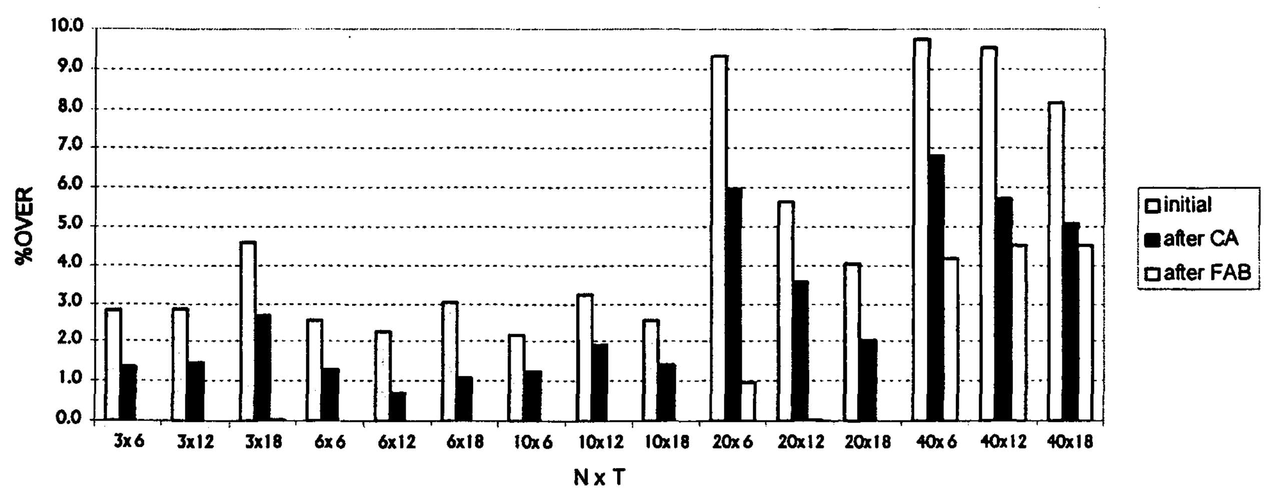

Figure 4. Infeasibility degree of the initial and heuristic solutions *versus* instance size.

In the following, the quality of the solutions generated by the heuristics FAB and CA is compared to a lower bound given by the Lagrangean relaxation with respect to constraints (3) and (4).

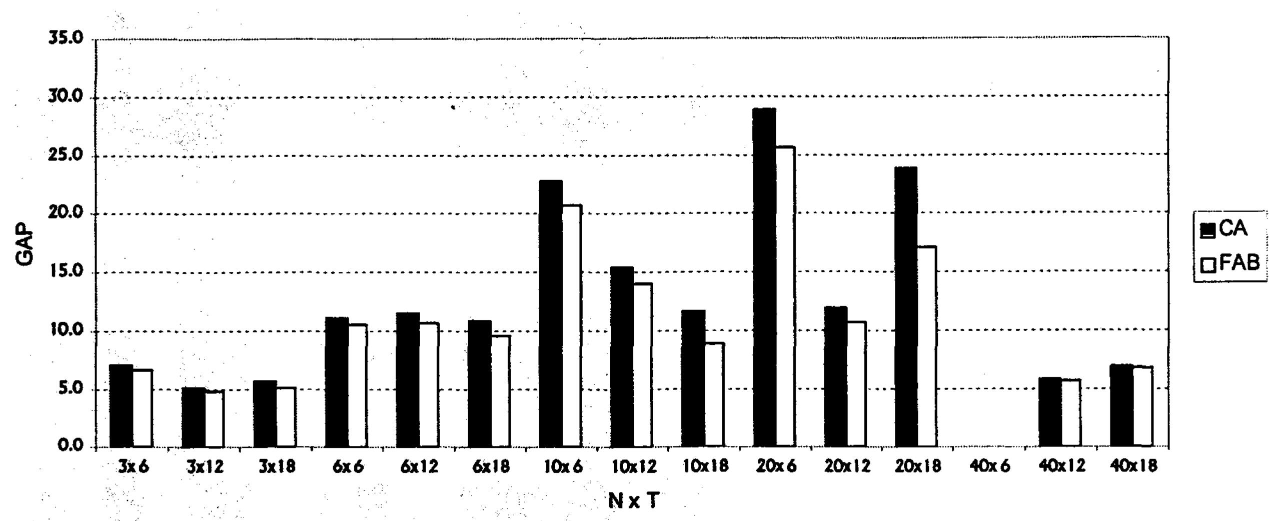

Figure 5. Gap between heuristic values and Lagrangean lower bound

The relaxed model derived from ME decomposes into a set of N independent, uncapacitated, single-item lot-sizing problem solvable by the Wagner-Whitin algorithm . Subgradient optimization is then used to obtain a lower bound. Figure 5 shows the average relative percentage deviation (GAP) between the heuristics and the lower bound for each instance size.

Table 10. Overall performance of the heuristics.

N		CA				FAB			
	OVER (WW)	FEA	GAP	Time(s)	OVER	FEA	GAP	Time(s)	OVER
3	3.4	42.8	6.1	0.00	1.84	99.4	5.6	0.00	0.01
6	2.6	38.9	11.2	0.00	1.04	100.0	10.4	0.01	0.00
10	2.7	25.3	19.7	0.00	1.53	99.4	17.8	0.03	0.00
20	6.3	2.5	21.8	0.00	3.87	79.7	16.6	0.10	0.34
40	9.1	2.5	6.8	0.01	5.86	31.4	6.7	0.35	4.39
Mean	**4.8**	**22.4**	**11.3**	**0.00**	**2.83**	**82.0**	**10.3**	**0.10**	**0.95**

Table 10 presents the performance of the heuristics with respect to feasibility and solutions' quality. Compared to heuristic CA, the heuristic FAB is able to obtain a much larger number of feasible solutions for all instance sizes while reducing the overuse of resources relative to the initial WW solution. Regarding to the quality measure GAP, the proposed heuristic performed better than heuristic CA, although not so significantly as compared to the feasibility improvement.

6. CONCLUSIONS

This article has proposed a new heuristic for lot-sizing in capacitated multi-stage serial systems. The mathematical model includes features, such as lead times and setup times, that can be found in practical applications and not often considered in the lot-sizing literature. The heuristic starts from a solution to the corresponding uncapacitated problem and attempts to obtain a feasible solution by shifting production amounts backwards and forwards in time. When these moves are not allowed, a new shift procedure (**create_shift**) which explores the dependency among components in the product structure is activated. Computational experiments involving 1800 instances with up to 40 components and 18 periods showed that the proposed heuristic was able to find feasible solutions in 84.4 % of the instances while reducing the overuse of resources relative to the initial WW solution. For instances with tight capacity (types 1 and 2) the maximum average infeasibility degree is less than 3% and for the other capacity levels this degree decreases to less than 1%.

ACKNOWLEDGMENTS

This work has been sponsored by the Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP) and the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

REFERENCES

AFENTAKIS, P., B. GAVISH, AND U. KARMAKAR (1984), "Computationally Efficient Optimal Solutions to the Lot-Sizing Problem in Multistage Assembly Systems," *Management Science*, 30, 2, 222-239.

BILLINGTON, P., J. BLACKBURN, J. MAES, R. MILLEN, AND L. N. VAN WASSENHOVE (1994), "Multi-Item Lotsizing in Capacitated Multi-Stage Serial Systems," *IIE Transactions*, 26, 2, 12-18.

BILLINGTON, P. J., J. O. MCCLAIN, AND L. J. THOMAS (1983), "Mathematical Programming Approaches to Capacity MRP Systems: Review, Formulation and Problem Reduction," *Management Science*, 29, 10, 1126-1141.

BILLINGTON, P. J., J. O. MCCLAIN, AND L. J. THOMAS (1986), "Heuristics for Multilevel Lot-Sizing with a Bottleneck," *Management Science*, 32, 8, 989-1006.

BLACKBURN, J. D. AND R. A. MILLEN (1982), "Improved Heuristics for Mutistage Requirements Planning Systems," *Management Science*, 28, 1, 44-56.

BLACKBURN, J. D. AND R. A. MILLEN (1984), "Simultaneous Lot-Sizing and Capacity Planning in Multi-Stage Assembly Processes," *European Journal of Operational Research*, 16, 84-93.

CHIU, H. N. AND T. M. LIN (1988), "An Optimal Lot-Sizing Model for Multi-Stage Series/Assembly Systems," *Computers and Operations Research*, 15, 5, 403-415.

CLARK, A. J. AND H. SCARF (1960), "Optimal Policies for a Multi-Echelon Inventory Problem," *Management Science*, 6, 475-490.

CLARK, A. R. AND V. A. ARMENTANO (1993), "Echelon Stock Formulation for Multi-Stage Lot-Sizing with Component Lead Times," *International Journal of Systems Sciences,* 24, 9, 1759-1775.

CLARK, A. R. AND V. A. ARMENTANO (1995), "A Heuristic Resource-Capacitated Multi-Stage Lot-Sizing Problem with Lead Times," *Journal of the Operational Research Society*, 46, 10, 1208-1222.

FLORIAN, M., J. K. LENSTRA, AND A. H. G. RINNOY KAN (1980), "Deterministic Production Planning Algorithms and Complexity," *Management Science*, 26, 7, 669-679.

FRANCA, P.M., V. A. ARMENTANO, R. E. BERRETTA, AND A. R. CLARK (1997), "A Heuristic Method for Lot-Sizing in Multi-Stage Systems", *Computers & Ops. Res.* 24, 9, 861-874.

HARRISON, T. P. AND H. S. LEWIS (1996), "Lot Sizing in Serial Assembly Systems with Multiple Constrained Resources," *Management Science*, 42, 1, 19-36.

KUIK, R. AND M. SALOMON (1990), "Multi-Level Lot-Sizing Problem: Evaluation of a Simulated Annealing Heuristic," *European Journal of Operational Research*, 45, 25-37.

KUIK, R., M. SALOMON, L. N. VAN WASSENHOVE, AND J. MAES (1993), "Linear Programming, Simulated Annealing and Tabu Search Heuristics for Lotsizing in Bottleneck Assembly Systems," *IIE Transactions,* 25, 1, 62-72.

MAES, J., J. O. MCCLAIN, AND L. N. VAN WASSENHOVE (1991), "Multilevel Capacitated LotSizing Complexity and LP based Heuristics," *European Journal of Operational Research*, 53, 131-148.

MAES, J. AND L. N. VAN WASSENHOVE (1991), "Capacitated Dynamic Lotsizing Heuristics for Serial Systems," *International Journal of Production Research*, 29, 6, 1235-1249.

ROLL, Y. AND R. KARNI (1991), "Multi-item, Multi-level Lot Sizing with an Aggregate Capacity Constraint," *European Journal of Operational Research*, 51, 73-87.

STEINBERG, E. AND H. A. NAPIER (1980), "Optimal Multi-Level Lot Sizing for Requirements Planning Systems," *Management Science,* 26, 12, 1258-1271.

TOKLU, B. AND J. M. WILSON (1992), "A Heuristic for Multi-Level Lot-Sizing Problems with a Bottleneck," *International Journal of Production Research*, 30, 4, 787-798.

VOLLMANN, T.E., W. T. BERRY, AND D. C. WHYBARK (1988), *Manufacturing Planning and Control Systems*, Second Edition, Dow Jones, Richard D. Irwin, Illinois.

WAGNER, H. M. AND T. M. WHITIN (1958), "Dynamic Version of the Economic Lot Size Model," *Management Science,* 5, 1, 89-96.