INT. J. PROD. RES., 1993, VOL. 31, NO. 1, 173–189

# A multi-family dynamic lot sizing problem with coordinated replenishments: a heuristic procedure

H. MURAT MERCAN† and S. SELCUK ERENGUC‡

Consider a manufacturing environment where multiple items are produced. These products are grouped into families due to their similarities in design and manufacturing. It pays off to coordinate the manufacture of products in a given family because preparing the facility for producing a family of products involves a major family setup. In addition each individual product in a given family may require a relatively minor setup. The objective is to determine the product lot sizes, over a finite planning horizon, that will minimize the total relevant cost, subject to demand and capacity constraints. In this paper we present an effective heuristic procedure for this problem. Computational results for the heuristic procedure are also reported. Our computational experience leads us to conclude that the heuristic procedure may be of considerable value as a decision making aid to production planners in a real-world setting.

## 1. Introduction

Single-level multi-item dynamic lot sizing problem with coordinated replenishments is the determination of lot sizes of several products over a planning horizon where the products share a common resource. In many manufacturing environments products may be classified into families where the setup structure makes coordinated replenishments attractive. In this type of lot sizing problem, there is a family (major) setup time when at least one item from a given family is produced in a period. An individual (minor) setup time may also be incurred when a product is produced in a period. The objective is to determine the time-phased production schedule that minimizes the total inventory carrying costs under demand and capacity constraints. We will refer to this type of problem as 'Multi-family dynamic lot sizing problem with coordinated replenishments' (MFDLC).

MFDLC is notable mainly for at least two reasons. First it is found in many manufacturing environments, where products are grouped together to take advantage of their similarities in manufacturing and design. That is, those items whose coordination results in cost savings and increased productivity are classified into one family. For example, one manufacturing environment where this kind of product grouping is seen is the group technology environment (Groover 1980).

Second, a good approach for solving MFDLC can provide insight to solving more general lot sizing problems. One such problem is the lot sizing of multiple items which share more than one resource.

Received March 1992.

† Bilkent University, College of Business Administration, 06533 Bilkent, Ankara, Turkey.
‡ University of Florida, College of Business Administration, Department of Decision and Information Sciences, Gainesville, FL 32611, USA.

0020-7543/93 $10·00 © 1993 Taylor & Francis Ltd.

MFDLC presented in this paper may be criticized since it does not consider setup costs. Most of the lot sizing models developed in the literature presuppose that there is a trade-off between setup cost and inventory holding cost. Therefore, these models suggest carrying inventory as long as total inventory cost is smaller than incurring new setup (see for example Tersine 1988). The underlying assumption in these models is that inventory is the lesser of the two evils. Therefore, these models result in larger lot sizes which in turn results in higher inventories. This philosophy, however, has been criticized both by practitioners and researchers. Zangwill (1987) claims that inventory is caused by inefficiency and views inventory as a barometer to indicate the effectiveness of other parts of the system. The just-in-time approach sees inventory as a waste (see for example Sugimori *et al.* 1977). The objective, as stated by Zangwill, is to cut inventory to the lowest possible level by elimimating the causes of inventory. By minimizing inventory holding costs, MFDLC tries to avoid carrying inventory unless it is absolutely necessary due to limited capacity. Moreover, in a production environment the setup cost consists of the labour cost due to setup time and certain out of pocket costs such as material wastage. However, in many situations (Pinto and Mabert 1986, Spencer 1980), a major portion, if not all, of the setup cost is the labour cost of the setup time. If one makes the increasingly popular assumption that labour costs are fixed in the short run (Pinto and Mabert 1986, Thompson 1983), then in those cases where all or a major portion of the setup cost is due to setup time, one can treat the setup cost as a fixed expense in the short run. Therefore, in the short-term production planning problem, setup cost ceases to be a relevant factor. Since the production planning problem we consider in this paper is primarily a short-term planning model, we consider setup as a factor in capacity absorption. Also as a consequence of the short-term nature of the model, we assume that variable production costs are constant over the planning horizon.

One can argue that MFDLC places no value on the time devoted to setups and, therefore, labour time could not be used for other purposes such as minor repairs and on-site quality control of the products. This argument is true only if workers are multifunctional and motivated to perform tasks other than their specific jobs. In the United States, however, workers are usually specialized and are not trained to do multiple jobs. To create a multifunctional workforce does not only require major changes in manufacturing philosophy but also requires extensive training. Nevertheless, the model presented in this paper does not necessarily assume specialization in the workplace. By properly selecting the setup time and capacity absorption rate, it is possible to incorporate minor repairs and quality control into MFDLC. For example, setup time may include time devoted to minor repairs. Similarly, the time necessary for on-site quality control may be included in the capacity absorption rate.

Erenguc and Mercan (1990) developed an exact branch and bound algorithm for solving MFDLC. Their computational experience demonstrated the need for developing effective heuristic procedures for solving MFDLC.

In this paper we develop a heuristic procedure for MFDLC. The heuristic procedure is tested with the benchmark problems which were provided by Erenguc and Mercan (1990).

This paper is organized as follows. In Section 2 we give the notation and the formulation of MFDLC. A brief review of previous work is given in Section 3. We describe the heuristic procedure in Section 4. Section 5 is devoted to computational results. In Section 6 we give some concluding remarks.

## 2. Statement of the problem

The kind of coordination we consider in this paper is as follows. There are $r$ distinct products which are classified into $m \leq r$ mutually exlusive and jointly exhaustive families. There is a family (major) setup time $S_i$, $i \in \{1, 2, \ldots, m]$ for each period $t \in \{1, 2, \ldots, T]$ when a positive amount of any product $j$ in family $i$ is produced. $S_i$ is the changeover time associated with converting the facility from the production of some other family to production within family $i$. There may also be an individual (minor) setup time $s_j$ associated with including item $j$ (of family $i$) in the replenishment of the family in period $t$. Therefore, $s_j$ represents the relatively minor setup time of switching to production of item $j$ from the production of some other item within the same family.

The objective is to determine the production schedule for each item in each period that minimizes the total cost of carrying inventory such that demand for each item in each period is satisfied without backlogging and the production capacity in each period is not exceeded. MFDLC can be formulated as an optimization problem. The reader is referred to Erenguc and Mercan (1990) for the optimization model. The heuristic procedure developed in this paper finds the lowest cost production schedule by ignoring the capacity constraint and then adjusts this schedule to achieve capacity feasible production batches. To explain the heuristic we need to define the capacity constraint. The following notation is given for this purpose:

- $T$ number of periods in the planning horizon
- $r$ number of products
- $I$ index set of all products, $I = \{1, 2, \ldots, r\}$
- $m$ number of families
- $I_i$ index set of products in family $i$, $i \in \{1, 2, \ldots, m\}$. For any $i \neq j$. $i, j \in \{1, 2, \ldots, m\}$, $I_i \cap I_j = \emptyset$ and $I_1 \cup I_2 \cup \ldots \cup I_m = I$
- $I_t^i$ index set of products in family $i$ in period $t$
- $r_i$ cardinality of $I_i$, $\sum_{i=1}^{m} r_i = r$

For each $t \in \{1, 2, \ldots, T]$ and each $j \in I$ and $i \in \{1, 2, \ldots, m]$, we define the following:

- $x_{jt}$ number of units of product $j$ produced in period $t$, $x_{jt} \geq 0$
- $y_{jt}$ ending inventory of product $j$ in period $t$, $y_{jt} \geq 0$
- $d_{jt}$ demand for product $j$ in period $t$
- $h_j$ inventory holding cost per unit per period for product $j$
- $B_t$ available capacity in period $t$
- $a_j$ capacity absorption rate per unit of product $j$
- $S_i$ family setup time incurred if a positive amount of any product $j \in I_i$ is produced, $S_i > 0$
- $s_j$ individual setup incurred if a positive amount of product $j$ is produced, $s_j \geq 0$

For each $i \in \{1, 2, \ldots, m\}$ and each $t \in \{1, 2, \ldots, T\}$, let $x_t^i = \{x_{jt} : j \in I_i\}$. Then for each $i \in \{1, 2, \ldots, m]$, $t \in \{1, 2, \ldots, T]$ and $x_t^i$, the joint setup time function, $G_{it}$, is given by

$$G_{it}(x_t^i) = \begin{cases} 0 & \text{if } \sum_{j \in Ii} x_{jt} = 0 \\ S_i & \text{if } \sum_{j \in Ii} x_{jt} > 0 \end{cases} \tag{1}$$

For each $j \in I$, $t \in \{1, 2, \ldots, T]$ and $x_{jt}$, the capacity absorption function $V_{jt}$ is defined as

$$V_{jt}(x_{jt}) = \begin{cases} 0 & \text{if } x_{jt} = 0 \\ s_j + a_j x_{jt} & \text{if } x_{jt} > 0 \end{cases} \tag{2}$$

For each $x_t^i$, the capacity absorption function $\tau_{it}$ is given by

$$\tau_{it}(x_t^i) = \left[ G_{it}(x_t^i) + \sum_{j \in I_i} V_{jt}(x_{jt}) \right]$$

We can state the capacity constraint in each period as follows:

$$\sum_{i=1}^{m} \tau_{it}(x_t^i) = \left[ G_{it}(x_t^i) + \sum_{j \in I_i} V_{jt}(x_{jt}) \right] \leq B_t \qquad \text{for all } t = 1, 2, \ldots, T \tag{3}$$

We make the following assumptions.

A. There can be no backlogging, demands should be satisfied in their respective periods of occurrence.
B. Replenishment lead times are negligible.
C. Inventories at the beginning and at the end of the planning horizon are zero.
D. A setup for product $j \in I$ is needed in period $t$, whether or not this product was produced in period $t-1$.
E. Setup costs are negligible.

Assumptions B and C can be relaxed without making any changes in the heuristic procedure. These assumptions are made for ease of presentation. The case of backlogging (i.e. assumption A) can be handled with some modifications in the heuristic procedure. Assumption D is needed due to the backward recursive nature of the heuristic procedure. Although this assumption seems to be restrictive, the solution obtained from the heuristic procedure would still be a feasible solution to the problem that would be obtained by relaxing this assumption. Also, we note that the parameters $c_j$, $h_j$, $s_j$, $a_j$ and $S_i$ are expected to be constant over the short run. However, they could be made time-varying without changing the heuristic procedure significantly.

## 3. Review of previous work

The setup structure studied in this paper has received considerable attention in the dynamic lot sizing literature. Early research on dynamic lot sizing models with family and individual setups introduced these setups only as a cost and assumed no capacity constraints. Therefore, previous work on dynamic lot sizing with family and individual setups is more suited for a procurement rather than a manufacturing environment. Research in the area of dynamic lot sizing models with family and individual setup costs is due to Erenguc (1988), Kao (1979), Spencer (1980), Veinott (1969) and Zangwill (1966). Although the coordinated replenishment models of Erenguc 1988, Kao (1979), Spencer (1980), Veinott (1969) and Zangwill (1966) assumed away the capacity constraints, the kind of capacity constraints we studied in this paper were alluded to in these references. For a detailed description of the algorithms developed in these papers the reader is referred to Erenguc (1988).

Mercan (1990), and Erenguc and Mercan (1990) proposed the mathematical formulation for MFDLC where the setup structure studied in this paper was introduced through capacity constraints. In addition, these authors developed an exact branch and bound algorithm for solving MFDLC.

**4. Description of the heuristic procedure**

We use the following obvious property of MFDLC to obtain a starting solution for the heuristic procedure.

*Property 1.* Let $(x', y') = \{x'_{jt} = d_{jt}, y'_{jt} = 0$ for all $j \in I$ and $t \in \{1, 2, \ldots, T\}\}$. If $(x', y')$ is feasible with respect to constraint (3) then $(x', y')$ is an optimal solution for MFDLC.

As stated in property 1, if for all $t \in \{1, 2, \ldots, T\}$ the capacity in period $t$ is sufficient to produce $x'_{jt}$ for all $j \in I$, then the optimal production schedule is obtained. In most cases however, such a production schedule will violate the capacity constraints in certain periods and therefore will not be feasible. The idea behind the heuristic procedure is to eliminate these capacity violations by shifting the production of certain items to earlier periods where there is excess capacity. For any production schedule $x = \{x_{jt}: j \in I, t \in \{1, 2, \ldots, T\}\}$, deficiency in a period is defined as the amount of capacity units by which the capacity constraint (3) of MFDLC for that period is violated.

Let $x'_{jt}, j \in \{1, 2, \ldots, r\}$ and $t \in \{1, 2, \ldots, T\}$, be defined as in property 1. If this schedule does not violate any of the constraint (3), then it is an optimal schedule. Therefore, the procedure terminates. However, usually there is at least one period $t$, $t > 1$, in which the capacity constraint is violated. Let $t' \in \{2, \ldots, T\}$ be the largest period index where there is deficiency; that is, for all $t > t'$, the above schedule satisfies constraint (3). Let $x^*$ denote the lowest cost schedule obtained by the heuristic. The heuristic procedure sets $x^*_{jt} = x'_{jt}, j \in I, t \in \{t'+1, t'+2, \ldots, T\}$. The deficiency in period $t'$ can be eliminated by shifting a part or all of the lot sizes of some products to period $t'-1$. We call 'a shift alternative', the choice of specific items along with their quantities to be shifted from period $t$ to period $t-1$. These shift alternatives will be explained in the subsequent sections. Since the quantity of items transferred from period $t$ to period $t-1$ represents the inventory carried from period $t-1$ to period $t$, the cost of a shift alternative can be computed easily. In period $t'$, first a set of shift alternatives are generated. Let $K(t')$ be the number of shift alternatives generated in period $t'$. The cost of the $K(t')$ alternatives are computed and ranked in ascending order. Without loss of generality, let $\{1, 2, \ldots, K(t')\}$ be the ordered set of the alternatives. Then from these $K(t')$ alternatives the first $k(t') \leq K(t')$ alternatives are chosen. The choice of $k(t')$ will be discussed in Section 5. Associated with each of the $k(t')$ alternatives is a partial (feasible) production schedule. Let $x(v, t')$ be the $v$th partial schedule in period $t'$. Then for each $v \in \{1, 2, \ldots, k(t')\}$, $x(v, t') = \{x_{jt}(v): j \in I, t \in \{t', t'+1, \ldots, T\}\}$ where $x_{jt'}(v)$ for all $j \in I$ is simply $x'_{jt}$, minus the number of units of item $j$ shifted to period $t'-l$ and for all $j \in I$ and $t \in \{t'+1, t'+2, \ldots, T\}$, $x_{jt}(v) = d_{jt}$. The cost of the partial schedule $x(v, t')$ is simply the sum of the cost of the inventory carried from period $t'-1$ to $t'$ and the cost of the partial schedule from which $v$ is generated. Let $C[(v, t'), (p, t'+1)]$ denote the cost of partial schedule $x(v, t')$. Here $p$ is the index of the shift alternative in period $t'+1$ from which shift alternative $v$ in period $t'$ is generated. Since there is no deficiency in period $t'+1$, the only shift alternative generated in period $t'+1$ would be not to shift any product from period $t'+1$ and therefore, $C[(1, t'+1), (1, t'+2)] = 0$. Associated with each of the $k(t')$ alternatives are the trial production quantities for period $t'-1$. Let $q_{jt'1}(v)$ be the trial production quantity in period $t'-1$ for each item $j \in I$ under the shift alternative $v \in \{1, 2, \ldots, k(t')\}$ from period $t'$. Then for each $j \in I$, $q_{jt'-1}(v)$ is simply $x'_{jt'-1}$ plus the quantity of item $j$ that was shifted to period $t'-1$ under the $v$th shift alternative in period $t'$.

For any $v \in \{1, 2, \ldots, k(t')\}$ for which the trial quantities $q_{jt'1}(v)$, $j \in I$ violate the capacity constraint of period $t'-1$, a number of shift alternatives which eliminate the infeasibility in period $t'-1$ are generated. Any $v\in\{1,2,\ldots,k(t')\}$ for which the trial quantities do not violate the capacity constraint in period $t'-1$, results in only one shift alternative in period $t'-1$. This alternative is not to shift any production to period $t'-2$. Let $K(t'-1)$ be the total number of shift alternatives generated in period $t'-1$. These shift alternatives are ranked in ascending order of their costs, $C[(v,t'-1),(p,t')]$ and the first $k(t'-1)\leqslant K(t'-1)$ where $v\in\{1,2,\ldots,k(t'-1)\}$ there is a partial feasible schedule $x(v,t'-1)=\{x_{jt}(v)\colon j\in I, t\in\{t'-1,t',\ldots,T\}\}$. This process is repeated until $t=2$. Note that shift operation is not applied in period 1, since it is the beginning of the planning horizon. To find a complete schedule, we set $x_{j1}(v)=q_{j1}(v)$ for all $j\in I$ and $v\in\{1,2,\ldots,K(2)\}$. If for any $v\in\{1,2,\ldots,K(2)\}$ trial quantities, $q_{i1}(v)$, $j\in I$, result in a deficiency in period 1, then this alternative is said to generate a *complete infeasible schedule*. Therefore, it is eliminated from consideration. The complete feasible schedule $x(v,1)=\{x_{jt}(v)\colon j\in I, t\in\{1,2,\ldots,T\}\}$ with the minimum cost $C[(v,2),(p,3)]$ among the $K(2)$ schedules is chosen as the 'best solution' generated by the heuristic procedure.

We should note that in each period where there is deficiency, the number of shift alternatives one can generate to eliminate the infeasibility is, in general, infinite. Since the objective is to minimize total inventory holding cost, deficiency in any period should be eliminated with the least possible inventory burden to the previous period. There are two general schemes for developing shift alternatives. We can eliminate the deficiency in any period by shifting the production of individual items, completely or partially, to the previous period without considering the family to which these items belong, or by shifting the production of all the items included in a family. In the first scheme, capacity savings for the period where there is deficiency, will come from smaller lot sizes for the shifted items and possibly from individual setup times. In the second scheme, capacity savings will result from family setup times in addition to individual setup times and processing times. Also note that, depending on the items that are shifted to the previous period, capacity savings from family setups may also be realized in the first scheme. The first scheme is referred to as *individual item release scheme* and the second scheme is referred to as *family release scheme*. Individual item release scheme and family release scheme are explained in the following sections.

To maintain ease of reading we relegate the mathematical development of the heuristic producudure to the appendices.

### 4.1. *Individual item release scheme*

The individual item release scheme considers the products independent of their families. In this approach, items are ranked in ascending order of their $h_j/a_j$ values. Note that $h_j/a_j$ is the holding incurred per unit saving in capacity usage. When two consecultive periods are considered, shifting the production of the item with the highest $h_j/a_j$ value is the best choice. Individual item release scheme is based on this idea. Let $\mu(w)$, $w\in\{1,2,\ldots,r]$ be the product index with the $w$th smallest $h_j/a_j$ ratio. For any period $t$, the $w$th alternative is generated in the following manner. The trial quantity of $\mu(w)$ is partially or completely shifted to period $t-1$. If this shift does not eliminate the deficiency in that period, the trial quantities of the remaining items are shifted in ascending order of their $h_j/a_j$ ratios until the deficiency is completely eliminated. We can generate $r$ different alternatives by using this scheme. Mathematical development of the individual item release scheme is given in Appendix B.

### 4.2. *Family release scheme*

The next method for generating shift alternatives is the family release scheme. In this scheme, a shift alternative is generated by sifting the total production of all the items in a set of families from period $t$ to period $t-1$. Let $B$ be a set of families whose production will be shifted from period $t$ to $t-1$. Note that $B$ may contain one or more families and the production quantities of all items whose families are in $B$ are shifted from period $t$ to period $t-1$. If there still remains some deficiency in period $t$ after the production quantities of all items whose families are in $B$ are shifted, the remaining deficiency is eliminated by shifting the production of individual items, starting with the item which has the lowest $h_j/a_j$ ratio and continuing to shift in ascending order of the $h_j/a_j$ ratios. Naturally, in this step individual items that belong to a family that is included in the set $B$ are bypassed. Mathematical development of the family release scheme is given in Appendix C.

Note that for each set $B$ only one shift alternative is generated. The number of shift alternatives that can be generated with the family release scheme depends on the number of different sets taken into consideration. We will comment on the possible choices of $B$ in discussing the computational results. In what follows we will first give a formal statement of the heuristic procedure and then give an example to illustrate the individual item and family release schemes. We will keep the formation statement as simple as possible. For a more mathematical treatment of the heuristic procedure the reader is referrred to the appendices.

### 4.3. *Formal statement of the heuristic procedure and an example*

Step 0 Initialization

- Set $t'=0$
- Determine $k(t)$ for all $t\in\{1,2,\ldots,T\}$ and the formation of the family release sets, i.e. the $B$ sets.
- Set $q_{jt}=d_{jt}$ for all $j\in I$, $t\in\{1,2,\ldots,T\}$ and determine the largest period index, $t'$, where there is a deficiency.
- If $t'=0$, set $x^*_{jt}$ for all $j\in I$ and $t\in\{1,2,\ldots,T\}$ and terminate the procedure. $x^*$ is the optimal solution for MFDLC.
- If $t'=T$, for all $j\in I$ set:
  $K(t'+1)=k(t'+1)=1$.
  $x_{jt'}(1)=d_{jt'}=x'_{jt'}$.
  $q_{jt'}(1)=d_{jt'}=x'_{jt'}$.
  $C[(1,t'),(1,t'+1)]=0$.
- If $t'<T$, for all $j\in I$ and $t'<t<T$, set:
  $K(t)=k(t)=1$.
  $x_{jt}(1)=d_{jt}=x'_{jt}$.
  $q_{jt}(1)=d_{jt}=x'_{jt}$.
  $C[(1,t),(1,t+1)]=0$.
- Set $t=t'$.

Step 1. For each of the $k(t+1)$ alternatives where the trial quantities $q_{jt}(v)$, $v\in\{1,2,\ldots,k(t+1)\}$, violate the capacity constraints generate shift alternatives using the individual item and family release schemes. Let $K(t)$ be the total number of shift alternatives generated in period $t$.

Step 2. For each of $v\in\{1,2,\ldots,K(t)\}$ determine the feasible partial schedule $x(v,t)=\{x_{js}(v): j\in, s\in\{t,t+1,\ldots,T\}\}$ by subtracting the units shifted to period $t-1$ from $q_{jt}(v)$.

Step 3. Determine the cost $C[(v,t), (p,t+1)]$ of each partial schedule $x(v,t)$, $v \in \{1,2,\ldots,K(t)\}$. This cost is the cost of carrying the inventories of the items shifted to period $t-1$ for one period plus the cost of the partial schedule $x(p,t+1)$ from which $x(v,t)$ follows.

Step 4. Rank the $K(t)$ alternatives in increasing order of their $C[(v,t), (p,t+1)]$ values. Select the first $k(t) \leq K(t)$ of these alternatives.

Step 5. Complete the trial quantities $q_{jt-1}(v)$ by adding the number of units shifted (from period $t$ to period $t-1$) to $d_{jt-1}$ under each of the $v \in \{1,2,\ldots,k(t)\}$ shift alternatives.

Step 6. Set $t=t-1$. If $t=1$ go to step 7, otherwise go to step 1.

Step 7. Set $x_{j1}(v)=q_{j1}(v)$ for all $j \in I$ and $v \in \{1,2,\ldots,K(2)\}$. Select the complete feasible schedule $x(v,1)=\{x_{jt}(v): j \in I, t \in \{1,2,\ldots,T\}\}$ with the minimum cost $C[(v,2), (p,3)]$ as the best heuristic solution. Terminate.

### 4.4. *Illustrative example*

Consider a four-period, six-item and two-family production planning problem. Relevant data for the problem are given in Tables 1 and 2. Items 1–3 are assumed to be in family 1 and items 4–6 are assumed to be in family 2. We will generate two alternatives in period 4 by using the individual item release scheme and one alternative in period 3 by using the family release scheme. This alternative in period 3 is generated from the first alternative of period 4.

| Item | Demand | | | | Other parameters | | |
|---|---|---|---|---|---|---|---|
| | Period 1 | Period 2 | Period 3 | Period 4 | Individual setup time | Capacity absorption rate | Holding cost (per unit per period) |
| 1 | 53 | 8 | 72 | 68 | 10 | 3 | 5·19 |
| 2 | 25 | 88 | 35 | 85 | 14 | 2 | 4·14 |
| 3 | 0 | 198 | 34 | 0 | 6 | 1 | 3·28 |
| 4 | 12 | 138 | 108 | 101 | 12 | 2 | 3·76 |
| 5 | 4 | 88 | 39 | 42 | 18 | 4 | 3·14 |
| 6 | 22 | 46 | 83 | 10 | 25 | 4 | 3·41 |
| Capacity | 1196 | 1875 | 1090 | 1094 | | | |

Table 1. Demand, individual setup time, capacity absorption rate, capacity data for a six-item four-period two-family problem.

| Family | Items in the family | Family setup time |
|---|---|---|
| 1 | 1,2,3 | 168 |
| 2 | 4,5,6 | 249 |

Table 2. Family setup time information.

*Two shift alternatives using the individual item release scheme*

We first set $x'_{jt} = d_{jt}$ for all $j \in I$ and $t \in \{1, 2, \ldots, T\}$. Then we compute $h_j/a_j$ for all $j \in I$. They are given as: $h_5/a_5 = 0{\cdot}79$, $h_6/a_6 = 0{\cdot}85$, $h_1/a_1 = 1{\cdot}73$, $h_4/a_4 = 1{\cdot}88$, $h_2/a_2 = 2{\cdot}07$, $h_3/a_3 = 3{\cdot}28$. The deficiency in period 4 is 186 units, which is computed from equation (3). Therefore, any shift alternative should save at least 186 units of capacity.

*Generation of the first shift alternative.* Since $w = 1$ and $\mu(1) = 5$, the first item to be released is 5. Total saving in period 4 as a result of this shift is 186 capacity units ($x'_{54}(1)(a_5) + s_1 = 168 + 18 = 186$). Therefore, item 5 will be completely shifted. No other product will be released since the deficiency is completely eliminated. We then compute the following quantities,

$$q_{53}(1) = 39 + 42 = 81;\ q_{j3}(1) - x'_{j3} \qquad \text{for all } j \in I,\ j \neq 5$$

$$x_{54}(1) = 0;\ x_{j4}(1) = x'_{j4} \qquad \text{for all } j \in I,\ j \neq 5$$

The cost of this alternative is $0{\cdot}0 + (42 \times 3{\cdot}14) = 131{\cdot}88$.

*Generation of the second shift alternative.* We will now generate the second alternative by using the individual item release scheme. Since $w = 2$, the first item to be shifted in this alternative is 6. This release will result in 65 units ($q_{64}(2)(a_6) + s_6 = 40 + 25$) of capacity savings. Remaining deficiency is 121 units. So item 5 will next be shifted. This deficiency is eliminated by shifting 30·25 units (121/4) of item 5. We then compute the following quantities, $q_{63}(2) = 83 + 10 = 93$; $q_{53}(2) = 39 + 30{\cdot}25 = 69{\cdot}25$; $q_{j3}(2) = x'_{j3}$ for all $j \in I, j \neq 5$ and 6. $x_{64}(2) = 0{\cdot}0$; $x_{54}(2) = 42 - 30{\cdot}25 = 11{\cdot}75$; $x_{j4}(2) = x'_{j4}$ for all $j \in I, j \neq 5$ and 6. The cumulative cost of this alternative is computed as follows:

$$C[(2, 4), (1, 5)] = 0{\cdot}0 + (10 \times 3{\cdot}41) + (30{\cdot}25 \times 3{\cdot}14) = 129{\cdot}93.$$

*A shift alternative using the family release scheme*

For the purpose of illustration we will generate an alternative in period 3 by using the family release scheme. Suppose that this alternative is generated from the first alternative of period 4. Trial quantities for all items in period 3 corresponding to the first alternative are given above. The cost of the first alternative in period 4 was computed as 131·88. Total deficiency in period 3 for the first alternative of period 4 is 592 capacity units. Assume that $B$ contains family one. Therefore, we will shift all the production of items 1–3 from period 3 to period 2. The total savings in capacity units in this case is 518 units. The remaining deficiency is 74 units. To eliminate this deficiency we will use the individual item release scheme. Since $\mu(1) = 5$ and item 5 is not in family 1, it will be shifted next. The amount to be shifted from 5 is 18·5 units (74/4). We then compute the following quantities, $q_{12}(1) = 8 + 72 = 80$; $q_{22}(1) = 88 + 35 = 123$; $q_{32}(1) = 138 + 108 = 246$; $q_{23}(1) = 88 + 18{\cdot}5 = 106{\cdot}5$; $q_{j2}(1) = x'_{j2}$ for all $j \in \{4, 6\}$; $x_{13}(1) = 0{\cdot}0$; $x_{23}(1) = 0{\cdot}0$; $x_{33}(1) = 0{\cdot}0$; $x_{53}(1) = 81 - 18{\cdot}5 = 62{\cdot}5$; $x_{j3}(1) = x'_{j3}$ for all $j \in \{4, 6\}$.

Cumulative cost of this alternative is computed as follows:

$$C[(1, 3), (1, 4)] = 131{\cdot}88 + (72 \times 5{\cdot}19 + 35 \times 4{\cdot}14 + 34 \times 3{\cdot}28 + 18{\cdot}50 \times 3{\cdot}14) = 820{\cdot}07$$

## 5. Computational experience with the heuristic procedure for MFDLC

In implementing the heuristic procedure for MFDLC, in each period we generated at most $r$ alternatives using the individual item release scheme as explained in Section 4.1. We also generated alternatives using the family release scheme as explained in Section 4.2. In implementing the family release scheme, we considered shifting the total production of all possible combinations of one and two families from period $t$ to period $t-1$. Therefore, the maximum number of alternatives generated in period $t$ for each alternative of period $t+1$ is less than or equal to $(m^2+m+2r)/2$. It should be noted that in a given period $t$, a certain family release application and a certain individual item release application may generate the same shift alternative. In the implementation of the heuristic procedure, such duplicate occurrences are eliminated. It is also possible that the individual item release scheme may generate a shift alternative by only partially shifting the items in a given family. This shift alternative is obviously less costly than the shift alternative that the family release scheme would generate by completely shifting the trial quantities of all the items in the same family. The latter alternative is clearly *inferior* to the former. All such inferior alternatives are eliminated from consideration.

For each period $t$, $t\in\{2,3,\ldots,T\}$, let $K(t)$ be the total number of shift alternatives generated. These shift alternatives were first ranked in ascending order of their costs and then, as discussed earlier, the least costly $k(t)$ of these $K(t)$, $(k(t)\leqslant K(t))$ alternatives were selected to proceed to period $t-1$. In the implementation $k(t)$, $t=\{2,\ldots,T\}$ was set equal to 10 and then to 20. For all the test problem, the heuristic procedure produced the same solution with $k(t)=10$ as with $k(t)=20$. Accordingly, our CPU times are reported with $k(t)=10$ for all $t\in\{1,2,\ldots,T\}$.

The heuristic procedure for MFDLC was tested on 50 sets of problems whose optimal solutions were provided by Erenguc and Mercan (1990). Each set contained five problems. Therefore, the procedure was tested on 250 benchmark problems. In the first 15 sets of problems, individual setup time $s_j=0$ for all $j\in I$ and the family setup time $S_i>0$ for all $i\in\{1,2,\ldots,m\}$. These problems have prefix 'A'. In the second 15 sets of problems, whose prefix is 'B', both $s_j$ and $S_i$ values are positive. It is noted that these problems are rather 'difficult' in the sense that they are 'capacity tight' and have at least several periods where there is not enough capacity to satisfy all the demand. The problem sets with prefixes 'D' and 'E' were obtained from sets A and B, respectively by increasing each period's capacity by 10%. The heuristic procedure was coded in FORTRAN and run on an IBM 3090/400.

The heuristic procedure obtained an optimal solution for all the problems with prefix D. Computational effort required to solve the problems with prefix D ranges between 3·70 and 4·05 CPU s with an average of 3·85 CPU s. For the same problems computational effort required by the branch and bound algorithm ranges between 4·2 and 133·8 s with an average of 10·6 s.

A summary of our computational results with the heuristic procedure for the problem sets with prefixes A, B and E are reported in Tables 3–5, respectively. We use the notation $P.x.y.z$ in the first column of Tables 3–5. The first letter indicates the prefix of the problem set and $x=r$, $y=T$ and $z=m$.

The heuristic procedure was able to generate a feasible solution for all the test problems. The average CPU time required by the heuristic procedure over all the test problems was 3·90 s, whereas the average CPU time required by the branch and bound algorithm for the same problems was reported as 122 s. A comparison of the computational effort for the branch and bound algorithm and the heuristic procedure

| Problem number | Difference between optimal and heuristic solution (%) | | | Number of problem optimal solutions found by heuristic | Average CPU time (s) |
|---|---|---|---|---|---|
| | Average | Minimum | Maximum | | |
| A.4.6.2 | 0·12 | 0·00 | 0·63 | 4 | 3·80 |
| A.4.8.2 | 0·48 | 0·00 | 1·24 | 3 | 4·50 |
| A.6.6.2 | 0·00 | 0·00 | 0·00 | 5 | 4·47 |
| A.6.6.3 | 0·20 | 0·00 | 9·95 | 3 | 3·85 |
| A.6.8.2 | 1·52 | 0·00 | 6·60 | 3 | 3·85 |
| A.6.8.3 | 1·15 | 0·00 | 5·75 | 4 | 3·86 |
| A.8.6.2 | 0·00 | 0·00 | 0·00 | 5 | 3·87 |
| A.8.6.4 | 0·03 | 0·00 | 0·17 | 4 | 3·88 |
| A.8.8.2 | 0·00 | 0·00 | 0·00 | 5 | 3·95 |
| A.8.8.4 | 1·76 | 0·00 | 4·60 | 1 | 4·05 |
| A.4.10·2 | 2·02 | 0·00 | 7·30 | 1 | 3·78 |
| A.6.10.2 | 1·46 | 0·00 | 7·30 | 4 | 3·85 |
| A.6.10.3 | 2·40 | 0·00 | 8·40 | 1 | 3·88 |
| A.8.10·2 | 0·34 | 0·00 | 1·70 | 4 | 3·98 |
| A.8·10·4 | 6·65 | 0·00 | 12·50 | 1 | 4·04 |

Table 3. Summary of computational results for MFDLC problems with zero individual setup time.

| Problem number | Difference between optimal and heuristic solution (%) | | | Number of problem optimum solutions found by heuristic | Average CPU time (s) |
|---|---|---|---|---|---|
| | Average | Minimum | Maximum | | |
| B.4.6.2 | 0·00 | 0·00 | 0·00 | 5 | 3·70 |
| B.4.8.2 | 0·56 | 0·00 | 1·40 | 2 | 3·77 |
| B.6.6.2 | 0·37 | 0·00 | 1·38 | 3 | 3·82 |
| B.6.6.3 | 5·16 | 0·00 | 20·05 | 3 | 3·83 |
| B.6.8.2 | 0·00 | 0·00 | 0·00 | 5 | 3·83 |
| B.6.8.3 | 1·89 | 0·00 | 6·93 | 1 | 3·85 |
| B.8.6.2 | 0·00 | 0·00 | 0·00 | 5 | 3·90 |
| B.8.6.4 | 2·04 | 0·00 | 9·00 | 3 | 3·91 |
| B.8.8.2 | 0·30 | 0·00 | 1·49 | 4 | 3·91 |
| B.8.8.4 | 6·90 | 0·00 | 23·00 | 1 | 3·95 |
| B.4.10.2 | 1·90 | 0·24 | 7·88 | 0 | 3·79 |
| B.6.10.2 | 0·23 | 0·00 | 1·15 | 4 | 3·86 |
| B.6.10.3 | 1·41 | 0·00 | 7·03 | 4 | 3·87 |
| B.8.10.2 | 0·00 | 0·00 | 0·00 | 2 | 4·02 |
| B.8.10.4[1] | 7·92 | 10·35 | 10·35 | 0 | 4·05 |

[1] Optimal solutions for only two problems were available in this set.

Table 4. Summary of computational results for MFDLC problems with positive individual setup times.

| Problem number | Difference between optimal and heuristic solution (%) | | | Number of problem optimal solutions found by heuristic | Average CPU time (s) |
|---|---|---|---|---|---|
| | Average | Minimum | Maximum | | |
| E.6.8.3 | 0·65 | 0·00 | 2·81 | 3 | 3·85 |
| E.8.6.2 | 0·00 | 0·00 | 0·00 | 5 | 3·90 |
| E.8.6.4 | 0·01 | 0·00 | 0·06 | 4 | 3·91 |
| E.8.8.2 | 0·00 | 0·00 | 0·00 | 5 | 3·91 |
| E.8.8.4 | 0·01 | 0·00 | 0·06 | 4 | 3·95 |
| E.4.10.2 | 0·00 | 0·00 | 0·00 | 5 | 3·79 |
| E.6.10.2 | 0·00 | 0·00 | 0·02 | 4 | 3·86 |
| E.6.10.3 | 0·01 | 0·00 | 0·03 | 4 | 3·87 |
| E.8.10.2 | 0·01 | 0·00 | 0·00 | 5 | 4·02 |
| E.8.10.4 | 0·00 | 0·00 | 0·47 | 3 | 4·05 |

Table 5. Summary of computational results for MFDLC problems with 10% more capacity.

| Prefix of the problem type | Computational effort for the branch and bound algorithm (CPU s) | | | Computational effort for the heuristic procedure (CPU s) | | |
|---|---|---|---|---|---|---|
| | Min. | Max. | Ave. | Min. | Max. | Ave. |
| A | 4·0 | 2760·6 | 90·0 | 3·78 | 4·50 | 3·95 |
| B | 4·2 | 3150·0 | 267·8 | 3·70 | 4·05 | 3·86 |
| D | 4·2 | 133·8 | 10·6 | 3·70 | 4·05 | 3·85 |
| E | 4·1 | 1180·2 | 119·1 | 3·79 | 4·05 | 3·93 |

Table 6. Summary of computational effort of branch and bound algorithm and the heuristic procedure.

is given in Table 6. We would also like to note that the branch and bound algorithm was not able to solve three of the problems in problem set B.8.10.4 in 3200 CPU s. However, the heuristic procedure took only 4·05 CPU s to find a solution for each of the three problems. Furthermore, these solutions had objective values that were smaller or equal to the upper bounds found by the branch and bound algorithm at the end of 3200 CPUs.

Heuristic solutions to 64, 58, 100 and 86% of the problems in sets A, B, D and E, respectively were optimal. The heuristic procedure obtained an optimal solution for 74% of all the problems solved. Average percentage differences between the heuristic solutions and the optimal solutions are 1·16, 1·91, 0·0, and 0·08%, for problem sets A, B, D and E, respectively. Overall average error of the solutions obtained by the heuristic procedure is 0·75%. In addition we made the following observations.

(1) Computational effort required by the heuristic procedure ranges between 3·70 and 4·47 CPU s, whereas computational effort required by the branch and bound procedure ranges between 4·0 and 31150 CPU s.
(2) For fixed $T$ and $r$, percentage difference between the optimal solution and the heuristic solution increases with $m$, in general. For example compare problem sets A.6.6.2 and A.6.6.3 in Table 4, B.6.6.2 and B.6.6.3 in Table 5, and E.8.10.2 and E.8.10.4 in Table 6.

(3) For fixed $m$ and $r$, as $T$ increases, percentage difference between the optimal solution and the heuristic solution increases in general. For example compare problem sets A.4.6.2 and A.4.8.2 in Table 4, B.4.6.2 and B.4.8.2 in Table 5, and E.8.8.4 and E.8.10.4 in Table 6.

(4) For fixed $m$ and $T$, percentage difference between the optimal solution and the heuristic solution tends to decrease with $r$.

Erenguc and Mercan (1990) report that obtaining optimal solutions for problem sets A and B require considerably more computing effort than problem sets D and E. The difficulty in solving the problems in sets A and B is attributed to the fact that these problems are 'capacity tight' problems. It is evident from Tables 3–5 and from the discussion above, that the heuristic procedure performed extremely well for the problem sets D and E, and its performance on the harder problems (sets A and B) was very satisfactory. The average CPU time required by the heuristic procedure is only a small fraction (about 4%) of that required by the exact branch and bound algorithm. Problem size did not seem to have a significant impact on the CPU time requirements of the heuristic procedure, for the test problems solved. However, for the same set of test problems, the CPU time requirements of the branch and bound algorithm increased dramatically with the problem size.

Considering the performance characteristics of the heuristic procedure that were discussed above and the fact that in many manufacturing environments capacity is usually not as tight as in the test problems, one is led to conclude that the heuristic procedure presented in this paper is of considerable value as a decision making aid to production planners in a practical setting.

## 6. Concluding remarks

In this paper we presented a heuristic procedure for a multi-family dynamic lot sizing problem with coordinated replenishments. In a manufacturing environment where multiple items are produced, these items are grouped into families due to their similarities in design and manufacturing. Preparing the manufacturing facility for the production of a family of items involves a major family setup time and, depending on the minor dissimilarities among the members of a product family, each individual item may also require a minor setup time. The objective is to determine production lot sizes that will minimize the total inventory holding cost subject to demand and capacity constraints. The heuristic procedure developed in this paper starts with an infeasible production schedule and eliminates the infeasibilities by shifting the production to earlier periods.

For the lot sizing problem considered in this paper, determining the optimal lot sizes which minimize the inventory holding cost over a finite planning horizon, is inherently a difficult problem. Previous research shows that determining optimal lot sizes takes excessive amounts of computational time even for problems of moderate dimensions. We would like to note that the procedure developed in this paper does not always guarantee a feasible solution. However, we were able to obtain a feasible solution for all the test problems solved. It is possible to develop sufficient feasibility conditions under cerain mild assumptions about the demand patterns. For example it was shown that the heuristic procedure always generates a feasible schedule provided that there is a positive demand for each item in each period (Mercan and Erenguc 1991). It was also shown that the procedure explained in this paper is capable of generating a feasible schedule if there is a positive demand for at least one item from each family in period 1 (Mercan and Erenguc 1991). The reader is referred to Mercan and Erenguc (1991) for these feasibility conditions.

The heuristic procedure was tested on a set of 250 benchmark problems whose optimal solutions are known. In a great majority of these problems the heuristic procedure yielded an optimal solution and in those cases where an optimal solution was not found, it produced a solution that was within very acceptable bounds. Furthermore the heuristic procedure has rather modest computational requirements. The branch and bound algorithm developed by Erenguc and Mercan does not restrict lot sizes to integer values. However, the heuristic procedure can be easily modified to restrict the production lot sizes to integer values only.

The heuristic procedure developed in this paper generates the shift alternatives in a period by evaluating the effects of these alternatives in the previous period. This is accomplished by ranking the items in order of their holding cost to capacity absorption rate. It is also possible to develop other 'shift criteria' which look back to several periods. The procedure can be modified to handle the MFDLC problems with backlogging. It is also possible to modify the procedure to solve lot sizing problems where major and minor setup costs along with setup times are relevant factors. We are currently working on this version of the heuristic procedure.

**Acknowledgment**

We would like to express our sincere thanks to two anonymous referees whose detailed comments helped improve the manuscript significantly.

**Appendix A**

We explain the mathematical development of the shift alternatives in this appendix. The following additional notation is given for this purpose. For any alternative $v \in \{1, 2 \dots, K(t)\}$ and $t \in \{2, \dots, T\}$:

$$L[(v,t),(p,t+1)] = \begin{cases} 1 & \text{if the } v\text{th shift alternative in period } t \text{ follows from the } p\text{th shift alternative in period } t+1 \text{ where } v \in \{1,2,\dots,K(t)\}, p \in \{1,2,\dots,k(t+1)\}, t \in \{2,3,\dots,t'\}. \\ 0 & \text{otherwise} \end{cases}$$

$C[(v,t),(p,t+)]$ cost of the partial production schedule $x(v,t)$ which followed from the $p$th shift alternative in period $t+1$, $v \in \{1,2,\dots,k(t)\}$, $p \in \{1,2,\dots,k(t+1)\}$, $t \in \{2,3,\dots,t'\}\}$.

$z_j(v,t)$ number of units of item $j$; $j \in I$ whose production was shifted from period $t$ to period $t-1$ under the $v$th shift alternative in period $t$.

$Q_t^0(v)$ the amount of deficiency (in units of capacity) in period $t$ under the shift alternative $v$.

$I_{it}^+(v)$ $= \{j \in I_i;\ q_{jt}(v) > 0\}$

$F_t(v)$ $= \{i \in \{1,2,\dots,m\}:\ I_{it}^+(v) \neq \emptyset]$

Then in period $t$ we have the following relation between $Q_t^0(v)$ and $q_{jt}(v)$ for all $j \in I$.

$$Q_t^0(v) = \max\left\{0, \left(\left[\sum_{i \in F_t(v)} S_i + \left(\sum_{j \in I_{it}^+(v)} [s_j + a_j q_{jt}(v)]\right)\right] - C_t\right)\right\} \tag{4}$$

Then for any shift alternative $v\in\{1,2,\ldots,K(t)\}$ and for any shift alternative $p\in\{1,2,\ldots,k(t+1)\}$, $C[(v,t),(p,t+1)]$ is given by

$$C[(v,t),(p,t+1)]=\begin{cases}\sum_{j\in I} h_j z_j(v,t)+C[(p,t+1),(w,t+2)] & \text{if } L[(v,t),(p,t+1)]=1\\ \infty & \text{otherwise}\end{cases}$$

where $w\in\{1,2,\ldots,k(t+2)\}$

To initiate the heuristic procedure, we set

$q_{jt}=d_{jt}$ for all $j\in I$ and $t\in\{1,2,\ldots,T\}$ and for all $j\in I$ and $t\in\{1,2,\ldots,T\}$

$K(t)=k(t)=1$

$L[(1,t),(1,t+1)]=1$

$C[(1,t),(1,t+1)]=0$

$x_{jt}(1)=d_{jt}=x^*_{jt}$

where the vector $x^*=\{x^*_{jt}\colon j\in I,\ t\in\{1,2,\ldots,T\}\}$ denotes the lowest cost schedule obtained by the heuristic procedure.

## Appendix B. Individual item release scheme

Let $w\in\{1,2,\ldots r\}$ be an alternative that will be generated in period $t$ by using individual item release scheme. We will refer to this alternative as $w$th *minimum alternative*. Let $v'$ be an alternative generated in period $t+1$ from which alternative $w$ in period $t$ is to follow. Let $k=1$ be the first attempt to eliminate a deficiency in period $t$ for alternative $w$. In the individual item release scheme, the first item whose production is shifted from period $t$ to period $t-1$ is $\mu(w)$. It is understood that $q_{\mu(w)t}(v')>0$. Assume that $\mu(w)\in I_i$. The values of $z_{\mu(w)}(w,t)$, $q_{\mu(w)t-1}(w)$ and $x(\mu(w),t)$ are given as

$$z_{\mu(w)}(w,t)=\begin{cases} q_{\mu(w)t}(v') & \text{if for some } j\in\{I_i\backslash\mu(w)\}q_{jt}(v')>0 \text{ and } Q_t^{k-1}(v')\geqslant A_1 \quad (5a)\\ & \text{or } A_3<Q_t^{k-1}(v')\leqslant A_1\\ q_{\mu(w)t}(v') & \text{if for all } j\in\{I_i\backslash\mu(w)\}q_{jt}(v')=0 \text{ and } Q_t^{k-1}(v')\geqslant A_2 \quad (5b)\\ & \text{or } A_3<Q_t^{k-1}(v')\leqslant A_2\\ Y & \text{otherwise}\end{cases}$$

where

$$A_1=q_{\mu(w)t}(v')a_{\mu(w)}+s_{\mu(w)}$$

$$A_2=q_{\mu(w)t}(v')a_{\mu(w)}+s_{\mu(w)}+S_i$$

$$A_3=q_{\mu(w)t}(v')a_{\mu(w)}$$

$$Y=\max\{0,\ Q_t^{k-1}(v')/a_{\mu(w)}\}$$

$$q_{\mu(w)t-1}(w)=x_{\mu(w)t-1}+z_{\mu(w)}(w,t) \quad (6)$$

$$x_{\mu(w)t}(w)=\max\{0,\ (q_{\mu(w)t}(w)-z_{\mu(w)}(w,t))\} \quad (7)$$

The remaining deficiency in period $t$ is given by $Q_t^k(v')$

$$Q_t^k(v') = \begin{cases} \max\{0, [Q_t^{k-1}(v') - (x_{\mu(w)t}(w)a_{\mu(w)} + s_{\mu(w)})]\} & \text{if condition in 5}c \text{ holds} \quad (8a) \\ \max\{0, [Q_t^{k-1}(v') - (x_{\mu(w)t}(w)a_{\mu(w)} + s_{\mu(w)} + S_i)]\} & \text{if condition in 5}b \text{ holds} \quad (8b) \\ 0 & \text{otherwise} \quad (8c) \end{cases}$$

where $Q_t^0(v')$ is computed from relation (4).

If $Q_t^k(v') = 0$, then the alternative generation scheme is completed. If $Q_t^k(v') > 0$, then additional items need to be shifted from period $t$. Suppose that $Q_t^k(v') > 0$. Then $k$ is set to $k+1$. Let the item to be shifted in the $k$th attempt be $\mu(b)$. Let $FL_k(w)$ be the index set of products whose proudction was shifted prior to the $k$th attempt under the $w$th shift alternative. Then $\mu(b)$ is given by

$$\mu(b) = \begin{cases} \mu(k) & \text{if } \mu(k) \notin FL_k(w) \\ \mu(k+1) & \text{otherwise} \end{cases}$$

The values of $z_{\mu(w)}(w, t)$, $q_{\mu(w)t-1}(w)$ and $x_{\mu(w)t}(w)$ are computed from equations (5), (6) and (7), respectively. Here, $\mu(w)$ in (5), (6) and (7) is replaced with $\mu(b)$. $Q_t^k(v')$ is computed from (8). This procedure is repeated until $Q_t^m(v') = 0$, where $1 \leqslant m \leqslant r$, that is until the deficiency is completely eliminated. Let the deficiency be completely eliminated in the $m$th attempt. Then we set

$$x_{\mu(w)t}(w) = q_{\mu(w),t}(v') \qquad \text{for all } w \geqslant m+1$$

$$z_{\mu(w)}(w, t) = 0 \qquad \text{for all } w \geqslant m+1$$

**Appendix C. Family release scheme**

For any $t \in \{1, 2, \ldots, T\}$ and any shift alternative $w$, we need to give the following additional notation.

$$I_{Bt}^+(w) = \{j : j \in I_i,\ i \in B, \quad \text{and} \quad q_{jt}(v) > 0\}$$

$$F_{Bt}(w) = \{i : \text{for some } j \in I_i,\ j \in I_{Bt}^+(w)\}$$

where $w$ is the shift alternative to be generated in period $t$ and $v$ is the shift alternative in period $t+1$ from which $w$ will follow.

We are now in a position to generate a shift alternative in period $t$ for set $B$ using the family release scheme. We will give the formulae for $z_j(w, t)$, $q_{jt-1}(w)$ and $x_{jt}(w)$. For all $j \in I_{Bt(w)}^+$ these formulae are given as follows:

$$z_j(w, t) = q_{jt}(v);\ q_{jt-1}(w) = z_j(w, t) + x_{jt-1};\ x_{jt}(w) = 0{\cdot}0$$

Let $k = 1$, and $Q_t^k(v)$ be the remaining deficiency after all the production of items whose families are in $B$ is shifted from period $t$ to period $t-1$. Then

$$Q_t^k(v) = \max\left\{0, Q_t^{k-1}(v) - \left(\sum_{i \in F_{Bt}(w)} \left[S_i + \sum_{j \in I_{Bt}^+(w)} s_j + q_{jt}(v)a_j\right]\right)\right\}$$

where $Q_t^0(v)$ is computed from relation (4).

If $Q_t^k(v) = 0$, $w$ has been generated. The procedure moves on to another set to generate another shift alternative. If $Q_t^k(v) > 0$, there still remains some deficiency. To eliminate the remaining deficiency, individual item release with the first minimum alternative scheme is applied.

## References

ERENGUC, S. S., 1988, Multiproduct dynamic lot-sizing model with coordinated replenishments. *Naval Research Logistics*, **35**, 1–22.

ERENGUC, S. S., and MERCAN, H. M., 1990, A multi-family dynamic lot sizing model with coordinated replenishments. *Naval Research Logistics*, **37**, 539–558.

GROOVER, M. P., 1980, *Automation Production Systems and Computer Aided Manufacturing* (Englewood Cliffs, NJ: Prentice-Hall).

KAO, E. P. C., 1979, A multi-product dynamic lot size model with individual and joint setup costs. *Operations Research*, **27**, 279–289.

MERCAN, H. M., 1990, Multi-family dynamic lot sizing problems with coordinated replenishments. Unpublished Dissertation Thesis, University of Florida, Gainesville, Florida.

MERCAN, H. M., and ERENGUC, S. S., 1991, A multi-family dynamic lot-sizing problem with coordinated replenishments: A heuristic procedure. Working Paper, Cleveland State University, January 1992.

PINTO, P. A., and MABERT, V. A., 1986, Joint lot-sizing rule for fixed labor cost situation. *Decision Sciences*, **17**, 139–150.

SPENCER, M. S., 1980, Scheduling components for group technology lines. *Production and Inventory Management*, **21**, 43–49.

SUGIMORI, V., KUSUNOKI, K., CHO, F., and UCHIKAWA, S., 1977, Toyota production system and kanban system materialization of just-in-time and respect-for-human system. *International Journal of Production Research*, **15**, 553–564.

TERSINE, R. J., 1988, *Principles of Inventory and Materials Management*, 3rd edition (New York: North-Holland).

THOMPSON, K., 1983, MRPII in the repetitive manufacturing environment. *Production and Inventory Management*, **23**, 1–14.

VEINOTT, F. A., Jr., 1969, Minimum concave cost solution of Leontief substitution models of multi-facility inventory systems. *Operations Research*, **17**, 262–291.

ZANGWILL, W. I., 1966, A deterministic multiproduct, multifacility production and inventory model. *Operations Research*, **14**, 486–507.

ZANGWILL, W. I., 0000, From EOQ towards ZI. *Management Science*, **33**, 1209–1223.