A Lagrangean-based Heuristic for Dynamic Multilevel Multiitem Constrained Lotsizing with Setup Times

Horst Tempelmeier • Matthias Derstroff
Universität zu Köln, Seminar für Allgemeine Betriebswirtschaftslehre, Industriebetriebslehre und Produktionswirtschaft, Albertus Magnus-Platz, D-50923 Braunschweig, Germany
Technische Universität Braunschweig, Fachgebiet Produktionswirtschaft, Braunschweig, Germany

In this paper a heuristic approach for the dynamic multilevel multiitem lotsizing problem in general product structures with multiple constrained resources and setup times is proposed. With the help of Lagrangean relaxation the capacitated multilevel multiitem lotsizing problem is decomposed into several uncapacitated single-item lotsizing problems. From the solutions of these single-item problems lower bounds on the minimum objective function value are derived. Upper bounds are generated by means of a heuristic finite scheduling procedure. The quality of the approach is tested with reference to various problem groups of differing sizes.
(*Production Planning; MRP; Capacity Constraints; Multilevel Lotsizing; Setup Times*)

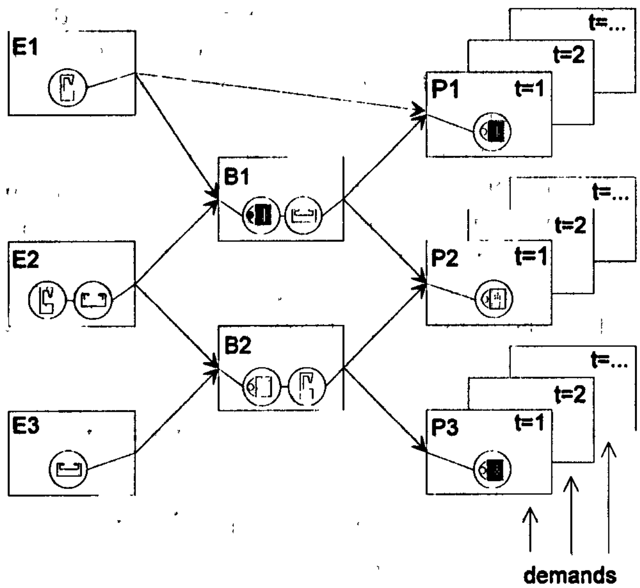
1. Introduction

This paper considers the *dynamic multilevel multiitem lotsizing problem for general product structures* and multiple capacity-constrained resources. This problem typically arises in one of the core planning steps in computerized Materials Requirements Planning (MRP) systems. It is well-known that MRP does not recognize capacity limitations in the lotsize planning phase, and hence normally overloads production facilities. The consequences are unpredictable and increased lead times and large work in process inventory. A lot of work has been devoted to the many types of lotsizing problems observable in industrial practice. For recent literature overviews see Bahl et al. (1987), Gupta et al. (1982), and Kuik et al. (1994). In the case of setup times even the problem of finding a feasible solution is NP-complete. One of the early approaches to multilevel capacitated lotsizing was developed by Billington (1983, 1986), who considers a single capacity-constrained resource. Maes (1987) proposed several LP-based heuristics that are applicable to assembly and linear product structures. In addition, he developed an approach based on a simple so-called ABC-heuristic (Maes and Van Wassenhove 1986, 1991), which is restricted to linear product structures. Pochet and Wolsey (1991) used a general purpose mathematical programming system with automatic cut generation routines to solve several variants of capacitated single-stage and multistage lotsizing problems. They emphasized the importance of further research on heuristics for problems with general product structure and capacity constraints. Recently Tempelmeier and Helber (1994, see also Helber 1994, 1995), developed a heuristic procedure that can be applied to general product structures as well as multiple resources required by products assigned to different levels in the bill of material structure. This procedure combines a cost modification approach (see Blackburn and Millen 1982, Heinrich and Schneeweiß 1986), with the Dixon-Silver (1981) procedure for single-level capacitated lotsizing.

In what follows, a multistep iterative procedure for the solution of the considered lotsizing problem is described. By means of *Lagrangean relaxation* of the *multilevel inventory balance equations* as well as of the *capacity constraints*, the problem is decomposed into several uncapacitated dynamic single-item lotsizing problems with time-varying production cost, which are solved to opti-

0025-1909/96/4205/0738$01.25

Figure 1 General Product/Operation Structure



mality. Violations of the inventory balance equations and the capacity constraints are taken into consideration implicitly through *Lagrangean multipliers*, which are updated by a subgradient optimization procedure. Based on the solutions of these single-item lotsizing problems a *lower bound* on the minimum value of the objective function is computed. *Upper bounds* on the objective function are computed by a heuristic finite loading procedure.

The approach taken in this paper compares most closely to the work of Billington. It differs from this author's work in several ways. Firstly, a true Lagrangean relaxation of the problem is used (Billington (1983) formulates a "heuristic" Lagrangean model). Secondly, sophisticated smoothing heuristics are applied to generate feasible solutions, including multiitem shifting of production from overloaded resources as well as several sequences of handling overloaded resources. Thirdly, computational tests are performed on general bill-of-material structures with several resources used by items assigned to different bill-of-material levels. As to our knowledge, currently there is no (exact or heuristic) lot sizing procedure available that covers all these aspects in an integrated way.

The plan of the paper is as follows: First, the problem considered is formulated as a mixed integer linear programming problem. Then the heuristic solution procedure is described. Finally, computational results are presented.

2. Problem Formulation

A graphical representation of the type of product/operation structure considered is depicted in Figure 1.

We consider several end items (denoted *Pn*), each with known external demands, over a finite horizon of discrete time buckets (shown by the cascaded rectangles). Several components and subassemblies (denoted *En* and *Bn*, respectively) are required for the production of the end items. A typical MRP system would consider the items associated to the rectangles as planning objects. This is possible in the MRP environment since the finite capacities of the resources are generally not considered. However, when finite capacity constrains lotsizes, then a more detailed perspective based on *operations* must be taken. Each operation, denoted by a circle, is assigned to a particular resource type. A specific resource type may be required to perform operations that are assigned to different levels of items in the bill-of-material structure.

As Figure 1 provides information not only about the input-output ("gozinto") structure, but also about the operations and resources required, we call this a *product/operation structure*. In the following we refer to the operations in terms of their results as items. Note that from the point of view of the resources a product/operation structure may be cyclic. This is the case, when a successor operation to be performed, say for end item *P*1, uses the same resource as a predecessor operation, say the first operation for subassembly *B*1.

The dynamic multilevel, multiitem capacitated lotsizing problem with multiple resources is stated as follows:

- The product structure is of the general type (with possibly several end products).
- Dynamic deterministic external demands for the end items are given.
- Several resources with given capacities are considered.
- Each operation is assigned to a single resource.
- Backlogging is not allowed.
- Once an item is produced, a fixed setup cost and a fixed setup time occur in addition to variable production cost.

• Inventory holding costs are computed based on the end-of-period inventory.

The mathematical programming formulation of the problem is as follows:

MODEL MLCLSP:

$$\text{Minimize } C = \sum_{t=1}^{T} \sum_{k=1}^{K} (p_{kt} \cdot q_{kt} + h_k \cdot y_{kt} + s_k \cdot \gamma_{kt}), \quad (1)$$

subject to

$$y_{k,t-1} + q_{k,t-z(k)} - \sum_{i \in N_k} a_{ki} \cdot q_{it} - y_{kt} = d_{kt}, \quad k = 1, 2, \ldots, K; \quad t = 1, 2, \ldots, T, \quad (2)$$

$$\sum_{k \in K_j} (tr_k \cdot \gamma_{kt} + tb_k \cdot q_{kt}) \leq b_{jt}, \quad j = 1, 2, \ldots, J; \quad t = 1, 2, \ldots, T, \quad (3)$$

$$q_{kt} - M \cdot \gamma_{kt} \leq 0, \quad k = 1, 2, \ldots, K; \quad t = 1, 2, \ldots, T, \quad (4)$$

$$q_{kt} \geq 0, \quad k = 1, 2, \ldots, K; \quad t = 1, 2, \ldots, T, \quad (5)$$

$$y_{kt} \geq 0, \quad k = 1, 2, \ldots, K; \quad t = 1, 2, \ldots, T, \quad (6)$$

$$\gamma_{kt} \in \{0, 1\}, \quad k = 1, 2, \ldots, K; \quad t = 1, 2, \ldots, T, \quad (7)$$

where (including notation used later)

- γ_{kt} binary setup variable for item k in period t
- a_{ki} production coefficient (number of units of item k required to produce one unit of item i)
- b_{jt} available capacity of resource j in period t
- d_{kt} external demand for item k in period t
- D_{kt} total requirement for item k in period t
- e_k echelon holding cost for item k
- h_k holding cost for item k
- J number of resources ($j = 1, 2, \ldots, J$)
- K number of items ($k = 1, 2, \ldots, K$)
- K_j set of indices of items that use resource j
- M large number
- N_k set of indices of the immediate successors of item k in the product/operation structure
- p_{kt} variable production cost for item k in period t
- q_{kt} lot size of item k in period t
- s_k setup cost for item k
- T number of periods ($t = 1, 2, \ldots, T$)
- tb_k production time per unit of item k
- tr_k setup time of item k
- u_{jt} Lagrangean multiplier for resource j in period t
- v_{kt} Lagrangean multiplier for item k in period t
- V_k set of indices of the immediate predecessors of item k in the product/operation structure
- y_{kt} inventory of item k at the end of period t
- $z(k)$ deterministic minimal lead time for item k

The objective function includes variable production costs, inventory holding costs, and setup costs. Constraints (2) for each item and period describe the relationship between the inventory level at the beginning and the end of the period, the external demand, the demand derived from production orders for successor items and the production quantity of an item. Constraints (3) enforce resource (capacity) limitations.

3. Description of the Heuristic Solution Method

Our solution method is based on the Lagrangean relaxation of the multi-level inventory balance constraints (2) and the capacity constraints (3). The overall procedure for the solution of model MLCLSP is outlined in Figure 2.

Through Lagrangean relaxation of the capacity constraints and the multilevel inventory balance constraints in Step 1 the capacitated multilevel, multiitem lotsizing problem is separated into several subproblems. The *relaxation of the capacity constraints* leads to an uncapacitated multilevel, multiitem lotsizing problem (for general product structures). Unfortunately up to now there is no efficient solution procedure available for this type of problem. Therefore a

Figure 2 **Outline of the Solution Procedure**

Step 0	Initialize Lagrangean multipliers.
Step 1	Update the *lower bound* on the minimum value of the objective function by solving a relaxed version of model MLCLSP, which consists of uncapacitated single-item Wagner-Whitin lotsizing problems.
Step 2	From the current solution compute resource *workloads* and *backorders* and *update the Lagrangean multipliers* with respect to the current workloads and backorders.
Step 3	Compute a new *upper bound* on the minimum value of the objective function; if a stopping criterion is not met, go to step 1; otherwise STOP.

second Lagrangean *relaxation with respect to the inventory balance constraints* is performed, which gives rise to single-item dynamic lotsizing subproblems of the Wagner-Whitin type, which can be solved efficiently. The solutions of these subproblems are used to compute an *actual lower bound* on the minimum value of the objective function.

In the following Step 2 first the resource workloads and the inventory levels resulting from Step 1 are determined with the help of Equations (2) and (3). The observed constraint violations are used to update the Lagrangean multipliers with a *subgradient optimization* procedure.

Finally, in Step 3 the actual upper bound on the minimum value of the objective function is computed. In order to avoid backorders the level-by-level requirements explosion procedure based on low level codes used in MRP systems is applied. Capacity constraints are taken into consideration through a heuristic finite loading procedure.

Computation of the Lower Bound

Lagrangean relaxation is a well-known way to circumvent the direct consideration of "difficult" constraints in an optimization model. This is done by multiplying the constraints with penalty costs and including them into the objective function (see e.g. Fisher 1981, Nemhauser and Wolsey 1988, Shapiro 1979). Lagrangean relaxation has been used for the solution of single-level (see Chen and Thizy 1990, Diaby et al. 1992, Fleischmann 1990, Lozano et al. 1991, Thizy 1991, Thizy and Van Wassenhove 1985, Trigeiro 1987, Trigeiro et al. 1989) as well as for multilevel lotsizing problems (see Billington 1983, Salomon 1991). In the current procedure the relaxation is performed with respect to the *capacity constraints* and to the *multilevel inventory balance constraints*. For ease of explanation it is assumed that all lead times $z(k)$ are zero. This issue will be taken up later on.

Relaxation of Model MLCLSP. In the following we aim at the formulation of uncapacitated dynamic single-item lotsizing problems. First the inventory variables y_{kt} are eliminated from model MLCLSP. Backorders of item k are impossible, if for each time interval between periods 1 and t the cumulated production quantity will be not less than the cumulated total requirement (external and derived demand) for that item. Assuming that beginning inventory is zero, the inventory of item k at the end of period t is given by Equation (8):

$$y_{kt} = \sum_{\tau=1}^{t} q_{k\tau} - \sum_{\tau=1}^{t} \left(d_{k\tau} + \sum_{i \in N_k} a_{ki} \cdot q_{i\tau} \right)$$
$$k = 1, 2, \ldots, K; \quad t = 1, 2, \ldots, T. \quad (8)$$

Substituting the variables y_{kt} in model MLCLSP results in the following reformulation (see Billington 1983, Salomon 1991).

MODEL MLCLSP_2.

Minimize C

$$= \sum_{t=1}^{T} \sum_{k=1}^{K} ((e_k \cdot (T - t + 1) + p_{kt}) \cdot q_{kt} + s_k \cdot \gamma_{kt}) - F \quad (9)$$

subject to

$$\sum_{\tau=1}^{t} q_{k\tau} \geq \sum_{\tau=1}^{t} \left(d_{k\tau} + \sum_{i \in N_k} a_{ki} \cdot q_{i\tau} \right),$$
$$k = 1, 2, \ldots, K; \quad t = 1, 2, \ldots, T, \quad (10)$$

$$\sum_{k \in K_j} (tr_k \cdot \gamma_{kt} + tb_k \cdot q_{kt}) \leq b_{jt},$$
$$j = 1, 2, \ldots, J; \quad t = 1, 2, \ldots, T, \quad (11)$$

$$q_{kt} - M \cdot \gamma_{kt} \leq 0,$$
$$k = 1, 2, \ldots, K; \quad t = 1, 2, \ldots, T, \quad (12)$$

$$q_{kt} \geq 0, \quad k = 1, 2, \ldots, K; \quad t = 1, 2, \ldots, T, \quad (13)$$

$$\gamma_{kt} \in \{0, 1\}, \quad k = 1, 2, \ldots, K; \quad t = 1, 2, \ldots, T. \quad (14)$$

Here F denotes a constant term with no influence on the structure of the solution.

When the Lagrangean relaxation of the capacity constraints using multipliers, u_{jt}, and of the multilevel inventory balance constraints using multipliers, v_{kt}, is done, omitting the constant term F and after some rearrangements for each item k we get a dynamic uncapacitated single-item lotsizing problem (SLULSP$_k$). As the multilevel inventory balance constraints are relaxed, the input-output relations between items (material flow between components or subassemblies and their parent items) are only implicitly accounted for through the Lagrangean multipliers. In order to guarantee nonnegative inventory levels we add inventory constraints to the relaxed single-item problems. These constraints (see Equations (16)) enforce that the cumulated

Figure 3 Heuristic Finite Loading Procedure

```
μ: = 1;                                                          {(„Inner") Iteration index}
while (μ ≤ μmax and SO > 0) do
  m: = 1;                                                        {First resource}
  while (m ≤ J and SO > 0) do
    if (μ is odd-numbered) then                                  {Forward scheduling phase}
      τ: = 1;
      while (τ < T and SO > 0) do
        choose resource j from Ms(m) according to position m;
        if (KNjτ > bjτ) then begin                               {Is resource overloaded?}
          perform SINGLE-ITEM forward-shifting and update SO;
          if (KNjτ > bjτ) then
            perform MULTI-ITEM forward-shifting and update SO;
          endif
        endif
        τ: = τ+1;
      endwhile;
    else begin                                                   {Backward scheduling phase}
      τ: = T;
      while (τ > 1 and SO > 0) do
        choose resource j from Mp(m) according to the position m;
        if (KNjτ > bjτ) then                                     {Is resource overloaded?}
          perform SINGLE-ITEM backward-shifting and update SO;
          if (KNjτ > bjτ) then
            perform MULTI-ITEM backward-shifting and update SO;
          endif
        endif
        τ: = τ−1;
      endwhile;
    endif;
    m: = m+1;                                                    {Next resource}
  endwhile;
  μ: = μ+1;                                                      {Next iteration}
endwhile;
```

production quantity of a component is sufficient to supply its parent items. The resulting uncapacitated single-item lot-sizing problem reads as follows:

MODEL SLULSP$_k$.

$$\text{Minimize } C_k = \sum_{t=1}^{T} (c_{kt} \cdot q_{kt} + s_{kt} \cdot \gamma_{kt}), \tag{15}$$

subject to

$$\sum_{\tau=1}^{t} q_{k\tau} \geq \sum_{\tau=1}^{t} D_{k\tau}, \quad k = 1, 2, \ldots, K; \quad t = 1, 2, \ldots, T, \tag{16}$$

and (12)–(14), where

$$c_{kt} = e_k \cdot (T - t + 1) + p_{kt} + \sum_{\tau=t}^{T} \left(\sum_{i \in V_k} a_{ik} \cdot v_{i\tau} - v_{k\tau} \right) + tb_k \cdot u_{jt}, \quad k = 1, 2, \ldots, K; \quad t = 1, 2, \ldots, T, \tag{17}$$

$$s_{kt} = s_k + tr_k \cdot u_{jt}, \quad k = 1, 2, \ldots, K; \quad t = 1, 2, \ldots, T, \tag{18}$$

$$D_{kt} = d_{kt} + \sum_{i \in N_k} a_{ki} \cdot D_{it}, \quad k = 1, 2, \ldots, K; \quad t = 1, 2, \ldots, T. \tag{19}$$

Problem SLULSP_k can be solved by means of one of the known procedures. Efficient implementations were developed by Federgruen und Tzur (1991) and Wagelmans et al. (1992). The lower bound on the minimum value of the objective function is given by:

$$\begin{aligned}\text{LB} = &\sum_{t=1}^{T}\sum_{k=1}^{K}(e_k\cdot(T-t+1)+p_{kt})\cdot q_{kt}+s_k\cdot\gamma_{kt}\\ &-\sum_{t=1}^{T}\sum_{k=1}^{K}v_{kt}\cdot\sum_{\tau=1}^{t}\left(q_{k\tau}-\sum_{i\in N_k}a_{ki}\cdot q_{i\tau}-d_{k\tau}\right)\\ &+\sum_{t=1}^{T}\sum_{j=1}^{J}u_{jt}\cdot\left(\sum_{k\in K_j}(tr_k\cdot\gamma_{kt}+tb_k\cdot q_{kt})-b_{jt}\right)-F.\end{aligned}\tag{20}$$

Updating of the Lagrangean Multipliers. Updating of the Lagrangean multipliers is done by *subgradient optimization* (see Nemhauser and Wolsey 1988). Given initial values u^0 and v^0, subgradient optimization generates a sequence of Lagrangean multipliers, u^l and v^l, through the addition of a direction vector which is multiplied by a step size λ^l. The *Lagrangean multipliers of the capacity constraints* are updated according to Equations (21),

$$u_{jt}^l=\max\left\{0;u_{jt}^{l-1}+\lambda^l\cdot\left(\sum_{k\in K_j}(tr_k\cdot\gamma_{kt}+tb_k\cdot q_{kt})-b_{jt}\right)\right\},\quad j=1,2,\ldots,J;\quad t=1,2,\ldots,T,\tag{21}$$

and the updating of the *Lagrangean multipliers of the inventory balance constraints* is done by Equations (22),

$$v_{kt}^l=\max\left\{0;v_{kt}^{l-1}+\lambda^l\times\left(-\sum_{\tau=1}^{t}\left(q_{k\tau}-\sum_{i\in N_k}a_{ki}\cdot q_{i\tau}-d_{k\tau}\right)\right)\right\},\quad k=1,2,\ldots,K;\quad t=1,2,\ldots,T.\tag{22}$$

The step size λ^l in iteration l is updated as follows:

$$\lambda^l=\delta^l\cdot\frac{(\text{UB}-\text{LB}(u^{l-1},v^{l-1}))}{\sqrt{\sum_{t=1}^{T}\sum_{j=1}^{J}\left\{\sum_{k\in K_j}[tr_k\cdot\gamma_{kt}+tb_k\cdot q_{kt}]-b_{jt}\right\}^2+\sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{\tau=1}^{t}\left(d_{k\tau}+\sum_{i\in N_k}a_{ki}\cdot q_{i\tau}-q_{k\tau}\right)^2}}.\tag{23}$$

The step size λ^l depends on the parameter δ^l, on the gap between the current lower bound $\text{LB}(u^{l-1}, v^{l-1})$ and the estimated minimum value of the objective function of the dual problem, which is approximated by the upper bound UB, and on the Euclidian norm of the deviations in the critical constraints (capacity and inventory balance constraints).

We update the parameter δ^l as follows. The initial value of δ^l is set to 2. δ^l is halved whenever $\text{LB}(u^{l-1}, v^{l-1})$ has not increased during N iterations (see Diaby et al. 1992, Fleischmann 1990). We tested differing values of N. Best results were obtained for $N = 4$. This updating scheme was found to perform best after extensive numerical comparisons with suggestions due to Held et al. (1974) and Shor (1968). In order to stabilize the convergence of the subgradients a smoothing procedure proposed by Crowder (1976) is used.

The procedure ends when one of the following *stopping criteria* is met:

(a) "Outer" Iteration limit.
(b) Maximum gap between upper and lower bound.
(c) Maximum violation of the critical constraints:

$$\sum_{t=1}^{T}\sum_{j=1}^{J}\left\{\sum_{k\in K_j}[tr_k\cdot\gamma_{kt}+tb_k\cdot q_{kt}]-b_{jt}\right\}^2+\sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{\tau=1}^{t}\left(d_{k\tau}+\sum_{i\in N_k}a_{ki}\cdot q_{i\tau}-q_{k\tau}\right)^2\le 0.0001.\tag{24}$$

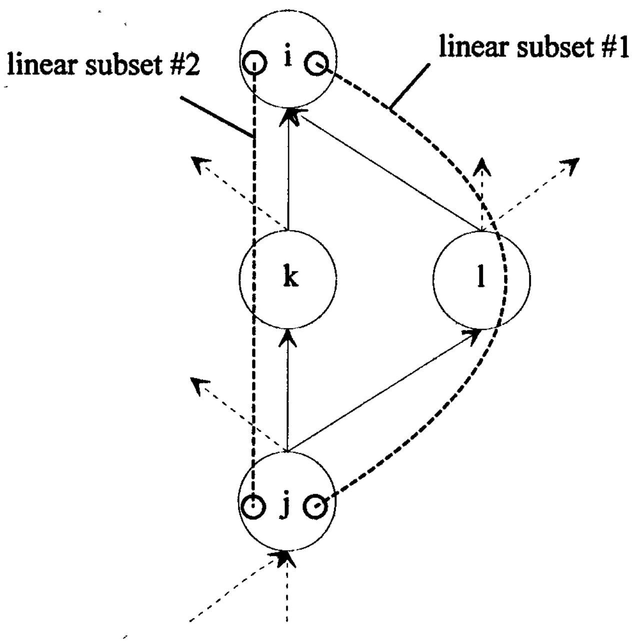
(d) Values of the Lagrangean multipliers:

$$u_{jt}\le 0.001\wedge v_{kt}\le 0.001,\quad j=1,2,\ldots,J;\quad k=1,2,\ldots,K;\quad t=1,2,\ldots,T.$$

Computation of the Upper Bounds

Inventory Balance Constraints. With respect to the inventory balance constraints it must be ensured that production quantities of upper-level items do not lead to backorders for the predecessor items. With respect

Figure 4 Linear Subsets between Item i and j



to this constraint, a feasible production can easily be constructed by planning lot sizes for the items according to their low-level code. First problem $SLULSP_k$ is solved for each end item. Then the planned production quantities are exploded down to the immediate predecessor level, where lot sizing is done next, etc. This is current MRP practice.

Capacity Constraints. The capacity constraints are taken into account through a heuristic finite scheduling procedure, which is applied to a production plan that is feasible with respect to the inventory balance constraints. First the actual workloads of the resources are computed and compared to the available capacities b_{jt} ($j = 1, 2, \ldots, J$; $t = 1, 2, \ldots, T$). A feasible solution of problem MLCLSP is then iteratively constructed by shifting production from periods with overloads to non-overloaded periods. This is done until a feasible solution has been reached or a given maximum number of finite scheduling iterations has been done. The finite scheduling procedure is made up of the following ingredients:

(a) *Sequence* of consideration of *periods* with capacity overloads

In a *forward scheduling* pass we start with the first period with capacity overloads and shift production quantities into future periods. In a *backward scheduling* pass we proceed vice versa. These scheduling variants are applied in an alternating way until a feasible solution without overloads has been found or until the given maximum number of finite scheduling ("inner") iterations has been reached.

(b) *Number* of *items* considered simultaneously

The shifting of production from an overloaded period to a different period may be done with respect to a single item (*single-item shifting*) or with respect to linearly connected subsets of the product/operation structure (*multiitem shifting*).

First it is tried to shift complete production orders of a single item, as this would not induce additional setups. Thereby we search for a period containing already a setup for the considered item. This would reduce the total number of setups. As this kind of testing is done with very low computational effort, it is used first to construct a feasible solution. However, if the single-item shifting fails, it is tried to simultaneously shift connected production quantities belonging to a linear segment of the product/operation structure. Thereby it is again tried to shift complete orders, if possible.

(c) *Sequence* of consideration of overloaded *resources*

The last decision that has to be made for the specification of the finite scheduling procedure refers to the

Figure 5 Example Product Structure

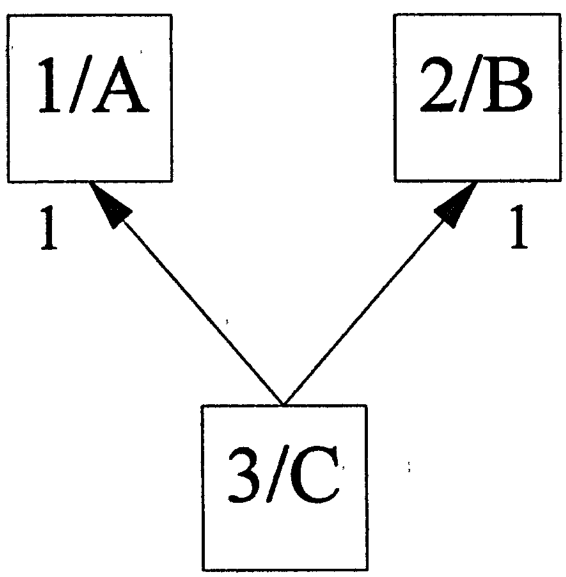


Table 1 Planning Instance 1

t	−1	0	1	2	3	4	5	6	7	8
d_{1t}			111	110	103	118	104	106	101	111
d_{2t}			166	152	148	156	125	116	139	153
y_{1t} (end of period)		**0**	0	46	118	0	0	37	111	0
y_{2t} (end of period)		**0**	0	20	47	66	116	0	0	0
q_{1t}			111	156	175	0	104	143	175	0
q_{2t}			166	172	175	175	175	0	139	153
Derived component demand			277	328	350	175	279	143	314	153
Available inventory (start of period)			292	365	525	175	422	143	467	153
y_{3t} (end of period)			15	37	175	0	143	0	153	0
q_{3t}	**292**	**350**	488	0	422	0	467	0	0	0

sequence, according to which overloaded resources are considered. In order to perform as many single-item shifts as possible, in the *forward scheduling phase* we start with considering overloaded resources required by items with few successors (at the beginning of the procedure these are the end items). Shifting production of an item with no or few successors forward (i.e., into a future period) reduces the possibility of inducing backorders for successor items.

In the *backward scheduling phase* we start with considering overloaded resources required by many low-level components (i.e., with few predecessors), as the production of these items can be shifted backward without inducing backorders at lower-level items.

The structure of the *finite loading heuristic* is presented in Figure 3, where the following symbols are used:

- μ iteration index
- $\mu_{\max}$ maximum number of iterations
- SO total overload on all resources (amount of infeasibility)
- KN_{jt} capacity requirement of resource j in period t
- τ period index
- j resource index
- M_S set of resources sorted according to minimum number of successors
- M_P set of resources sorted according to minimum number of predecessors

The production quantity to be shifted into an earlier or later period is determined as follows. Consider a resource j that is currently overloaded in period τ. This overload is reduced by a *single-item* shifting of production of item i into a target period t, whereby i and t are selected such that the maximum decrease or the minimum increase in the value of the objective function results.

In case of *forward shifting*, one has to bear in mind that shifting of production of item i into a later period $t > \tau$

Table 2 Planning Instance 2

t	2	3	4	5	6	7	8	9	10	11
d_{1t}			119	104	106	101	111	106	103	93
d_{2t}			156	125	116	139	153	131	154	139
y_{1t} (end of period)		**118**	35	106	0	63	127	21	93	0
y_{2t} (end of period)		**47**	0	0	0	0	0	0	0	0
q_{1t}			36	175	0	164	175	0	175	0
q_{2t}			109	125	116	139	153	131	154	139
Derived component demand			145	300	116	303	328	131	329	139
Available inventory (start of period)			175	452	152	303	459	131	468	0
y_{3t} (end of period)		**175**	30	152	36	0	131	0	139	0
q_{3t}	**0**	**422**	0	267	459	0	468	0	0	0

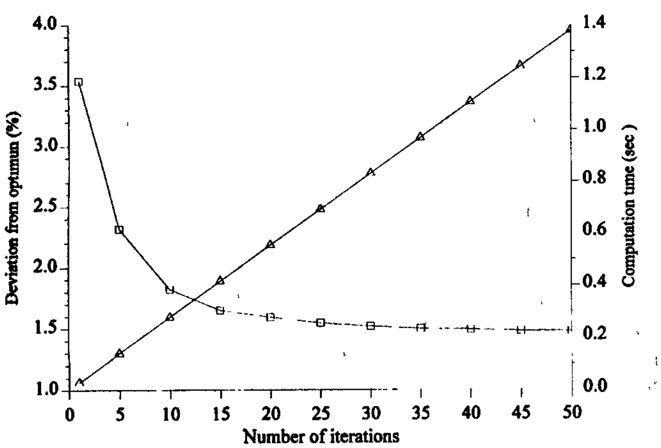
Table 3 Average Deviations from Optimality (Problem Class A)

		Utilization Profile					
TBO Profile	CV	90	70	50	90/70/50	50/70/90	mean
1	0.1	0.05%	0.00%	0.00%	0.00%	0.00%	0.01%
	0.4	1.09%	0.10%	0.01%	0.53%	0.66%	0.48%
	0.7	2.49%	0.97%	0.18%	2.25%	1.24%	1.43%
	mean	1.21%	0.36%	0.06%	0.93%	0.63%	0.64%
2	0.1	0.00%	1.59%	1.70%	0.75%	0.39%	0.88%
	0.4	1.52%	0.97%	0.75%	1.26%	1.11%	1.12%
	0.7	1.79%	1.17%	0.54%	1.70%	1.22%	1.28%
	mean	1.10%	1.24%	0.99%	1.24%	0.90%	1.10%
4	0.1	0.23%	5.38%	2.84%	1.52%	1.06%	2.21%
	0.4	2.00%	4.84%	1.76%	2.63%	1.57%	2.56%
	0.7	2.28%	3.76%	2.83%	2.76%	2.55%	2.84%
	mean	1.50%	4.66%	2.48%	2.30%	1.73%	2.54%
1/2/4	0.1	0.23%	0.29%	0.60%	0.55%	0.28%	0.39%
	0.4	2.62%	0.76%	0.80%	1.33%	0.37%	1.18%
	0.7	2.22%	1.17%	0.80%	1.69%	1.32%	1.44%
	mean	1.69%	0.74%	0.73%	1.19%	0.66%	1.00%
4/2/1	0.1	0.54%	1.75%	4.26%	0.93%	0.42%	1.58%
	0.4	1.49%	2.29%	1.83%	2.01%	0.73%	1.67%
	0.7	1.92%	5.75%	1.60%	2.95%	2.71%	2.99%
	mean	1.32%	3.27%	2.56%	1.96%	1.29%	2.08%
		overall mean (1500 problem instances)					1.47%

must not lead to backorders for this item resulting from successor-item demands. If the quantity $\Delta_{i\tau t}$ is shifted from period τ into the target period t, then it is no longer available to supply requirements during periods τ until $t-1$. Consequently in all periods from τ until $t-1$ inventory is reduced by $\Delta_{i\tau t}$. Therefore the maximum quantity of item i that may be shifted from period τ into period t (single-item shift), is given by:

$$\Delta_{i\tau t}^{\max} = \min(q_{i\tau}, \min_{\tau \le j \le t-1} \{y_{ij}\}), \quad t > \tau. \qquad (25)$$

Figure 6 Average Deviation from Optimality and Average Computation Time as a Function of the Number of Iterations (Problem Class A)



If there are no further items causing the overload of the considered resource, for which a single-item shifting could be performed, then we will proceed to the *multiitem shifting*. For all direct and indirect successors j of item i, for which the maximum shifting quantity according to Equation (25) is positive, the quantity is computed that must be shifted simultaneously with item i. Thereby only the items on *linear subsets* of the product/operation structure including item i and j are taken into account. In the example shown in Figure 4, two subsets—including items i, k, and j and items i, l, and j—are considered.

In case of *backward shifting*, production of item i is shifted from period τ into an earlier period $t < \tau$. In this case the consequences for the derived demands and

Table 4 Average Deviations from Optimality (Problem Class B)

		Utilization Profile					
TBO Profile	CV	89	68	48	89/68/49	49/69/88	mean
1	0.1	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
	0.4	0.74%	0.01%	0.00%	0.79%	0.33%	0.37%
	0.7	2.24%	0.24%	0.25%	1.24%	0.83%	0.96%
	mean	0.99%	0.08%	0.08%	0.68%	0.39%	0.44%
2	0.1	0.13%	1.10%	0.00%	0.48%	0.73%	0.49%
	0.4	1.39%	0.80%	0.49%	1.13%	0.19%	0.80%
	0.7	1.35%	0.78%	0.51%	1.51%	1.49%	1.13%
	mean	0.96%	0.89%	0.33%	1.04%	0.81%	0.81%
4	0.1	0.28%	4.88%	0.08%	2.55%	0.91%	1.74%
	0.4	2.83%	4.53%	2.59%	3.27%	1.13%	2.87%
	0.7	3.54%	1.99%	0.57%	3.24%	2.68%	2.40%
	mean	2.22%	3.80%	1.08%	3.02%	1.57%	2.34%
1/2/4	0.1	0.18%	0.86%	0.84%	0.96%	0.14%	0.60%
	0.4	3.05%	0.17%	0.91%	1.18%	0.36%	1.13%
	0.7	4.40%	0.58%	1.28%	1.63%	2.53%	2.09%
	mean	2.54%	0.54%	1.01%	1.26%	1.01%	1.27%
4/2/1	0.1	0.58%	2.31%	0.00%	0.82%	0.39%	0.82%
	0.4	1.46%	1.19%	1.55%	2.15%	0.12%	1.29%
	0.7	0.85%	4.71%	1.53%	3.21%	3.77%	2.81%
	mean	0.96%	2.74%	1.03%	2.06%	1.43%	1.64%
		overall mean (600 problem instances)					1.30%

the inventory of the immediate predecessors $j \in V_i$ are considered. The maximum quantity of item i that may be shifted from period τ into period t is given by:

$$\Delta_{i\tau t}^{\max} = \min\left(q_{i\tau}, \min_{j \in V_i \wedge t \leq l \leq \tau-1} \left\{\frac{y_{jl}}{a_{ji}}\right\}\right), \quad t < \tau. \quad (26)$$

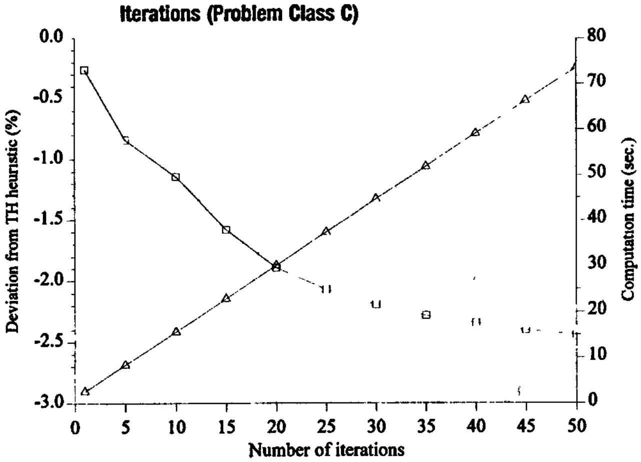
If for all items the maximum shiftable quantity is zero, then a multiitem shifting is performed.

Positive Minimal Lead Times and Initial Inventory. The problem under consideration typically arises in the context of a (computerized) production planning and control system, where lotsizing is performed in a rolling schedule mode, lead times are nonzero, and physical inventory and outstanding orders are passed from one planning cycle to the next. In order to coordinate a planning problem considered at time τ with its immediate predecessor (solved at time $\tau - R$, where R is the length of a planning cycle), for each item k it is necessary to include the lot sizes $q_{k,\tau-1}, q_{k,\tau-2}, \ldots, q_{k,\tau-z(k)}$ and the initial inventory $y_{k,\tau-1}$ resulting from the planning problem solved at time $\tau - R$ as data into the constraints (2). If the forecasted end item demands are the

Table 5 Average Deviations from the Tempelmeier/Helber Solution (Problem Class C)

Utilization Profile	1 Iteration	50 Iterations	TBO	1 Iteration	50 Iterations	CV	1 Iteration	50 Iterations
90	1.89	−0.83	1	−0.38	−0.58	0.1	−0.52	−2.36
70	−0.07	−2.57	3	−0.73	−3.17	0.5	−0.07	−2.42
50	−2.04	−3.64	5	−0.66	−4.35	0.9	−0.22	−2.56
90/70/50	−1.36	−3.22	1/3/5	−1.10	−4.41			
50/70/90	0.25	−1.97	5/3/1	1.54	0.28			
mean	−0.27	−2.45	mean	−0.27	−2.45	mean	−0.27	−2.45

Figure 7 **Average Deviation from the Tempelmeier/Helber Solution and Average Computation Time as a Function of the Number of Iterations (Problem Class C)**



same as at time $\tau - R$, then a feasible production schedule can be constructed, as enough production of the predecessor items will have been scheduled in advance (i.e., in periods $\tau - 1, \tau - 2, \cdots$).

If forecasted end item demands have been revised and increased compared to the quantities forecasted at time $\tau - R$, then the problem of infeasibility may arise. This will happen when the increase in end item demand induces an increase in derived demand that cannot be satisfied either through quantities on order or through initial inventory—even if enough capacity is available. In this situation one will be forced to use safety stocks.

The most interesting situation arises when forecasted end item demands have increased, but only to a small extent. In this case a feasible production plan can be constructed. However, the use of model MLCLSP with updated data compared to the last planning cycle may render a different allocation of initial inventory and outstanding orders. This will be discussed with the help of a small example. Consider the product structure depicted in Figure 5.

End item 1 (2) is produced by resource A (B) with a capacity of 175 (175). Component 3 is produced by resource C, which has a capacity of 500. Assume that the lead time of component 3 is $z(3) = 2$ periods $[z(1) = z(2) = 0]$. Model MLCLSP is applied in a rolling schedule mode with a plan-

Table 6 **Average Deviations of the Heuristic Solution from the Lower Bound (Problem Class C)**

		Utilization Profile					
TBO Profile	CV	90	70	50	90/70/50	50/70/90	mean
1	0.1	0.24%	0.00%	0.00%	0.14%	0.16%	0.11%
	0.5	19.40%	4.12%	0.58%	20.29%	4.74%	9.83%
	0.9	23.79%	12.10%	3.86%	34.29%	10.70%	16.95%
	mean	14.48%	5.41%	1.48%	18.24%	5.20%	8.96%
3	0.1	20.12%	13.94%	10.20%	12.90%	17.58%	14.95%
	0.5	25.80%	14.79%	9.75%	14.69%	17.05%	16.41%
	0.9	30.36%	15.52%	8.70%	19.77%	17.25%	18.32%
	mean	25.43%	14.75%	9.55%	15.79%	17.29%	16.56%
5	0.1	27.20%	21.13%	12.55%	18.85%	25.83%	21.11%
	0.5	33.03%	22.38%	13.65%	21.57%	24.71%	23.07%
	0.9	37.92%	21.35%	12.37%	24.55%	23.02%	23.84%
	mean	32.72%	21.62%	12.85%	21.66%	24.52%	22.67%
1/3/5	0.1	19.03%	11.24%	6.63%	8.44%	17.67%	12.60%
	0.5	23.08%	11.65%	6.42%	10.84%	15.95%	13.59%
	0.9	27.32%	14.22%	6.69%	15.79%	16.74%	16.15%
	mean	23.14%	12.37%	6.58%	11.69%	16.79%	14.11%
5/3/1	0.1	26.59%	18.24%	16.36%	18.57%	21.29%	20.21%
	0.5	32.68%	21.48%	15.85%	24.47%	21.74%	23.25%
	0.9	34.84%	23.27%	16.08%	28.15%	20.77%	24.62%
	mean	31.37%	21.00%	16.10%	23.73%	21.27%	22.69%
		overall mean (600 problem instances)					17.00%

Table 7 Average Deviations of the Heuristic Solutions from the Lower Bound (Problem Class D)

		Utilization Profile					
TBO Profile	CV	89	71	53	92/70/51	53/71/88	Mean
1	0.1	1.24%	23.68%	24.84%	0.00%	3.62%	10.68%
	0.5	11.63%	0.00%	0.23%	2.22%	0.50%	2.92%
	0.9	19.74%	24.14%	0.71%	5.89%	9.68%	12.03%
	mean	10.87%	15.94%	8.59%	2.70%	4.60%	8.54%
3	0.1	22.10%	20.44%	24.66%	13.92%	14.10%	19.04%
	0.5	13.66%	11.37%	10.64%	9.63%	14.17%	11.89%
	0.9	13.61%	16.80%	19.56%	15.83%	16.47%	16.46%
	mean	16.46%	16.20%	18.29%	13.12%	14.91%	15.80%
5	0.1	33.99%	28.04%	32.17%	19.43%	19.16%	26.56%
	0.5	18.54%	13.15%	14.23%	12.14%	21.88%	15.99%
	0.9	19.13%	19.66%	30.51%	21.94%	26.06%	23.46%
	mean	23.89%	20.29%	25.64%	17.84%	22.37%	22.00%
1/3/5	0.1	24.24%	20.27%	23.17%	10.48%	11.60%	17.95%
	0.5	11.96%	6.76%	7.19%	7.31%	8.71%	8.39%
	0.9	10.52%	14.73%	23.50%	14.67%	16.52%	15.99%
	mean	15.57%	13.92%	17.95%	10.82%	12.28%	14.11%
5/3/1	0.1	27.10%	26.91%	30.00%	19.15%	20.15%	24.66%
	0.5	21.75%	14.96%	16.62%	16.58%	24.19%	18.82%
	0.9	21.69%	24.77%	23.41%	20.49%	23.04%	22.68%
	mean	23.51%	22.21%	23.34%	18.74%	22.46%	22.05%
		overall mean (600 problem instances)					16.50%

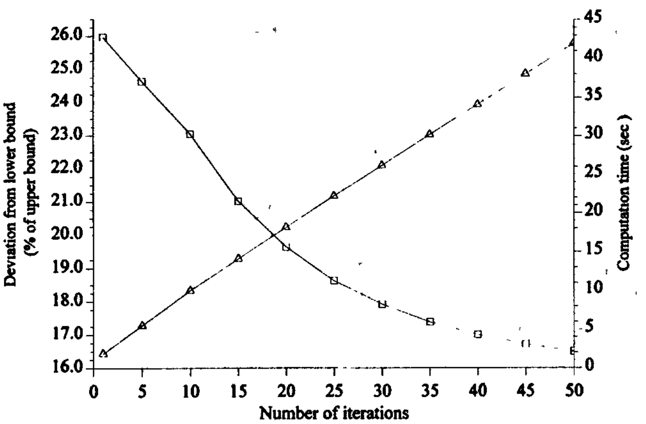
ning cycle of $R = 3$ periods and a planning window of $T = 8$ periods. The first MLCLSP instance is set up in period 0. Assume that for component 3 production quantities $q_{3,-1} = 292$ and $q_{3,0} = 350$ have been scheduled for periods -1 and 0, respectively. These time-phased on-order quantities are included as data into constraints (2) for component 3. Given certain cost parameters, the optimal production plan reads as shown in Table 1.

The planned available inventory of component 3 (= physical inventory + time-phased quantity on order) in periods 1 and 2 limits the total production quantities of items 1 and 2 in periods 1 and 2, respectively.

Now assume that the next planning instance is set up at the end of period 3. Note that end item 1 (2) inventory in period 3 is 118 (47) and the available inventory (on hand + on order) of the component at the beginning of period 4 is 175, and that at the beginning of period 5 additional 422 units will be available. Assume further that the forecasted demand for end item 1 in period 4 has changed from 118 to 119. In order to take account of the initial inventory on hand and the quantity on order for component 3, model MLCLSP is set up with constraints (2) including the outstanding order for item 3 produced in period 3 and available at the beginning of period 5 and including positive initial inventories. The optimal solution for the problem is shown in Table 2.

As long as the demand quantities between successive planning instances remain unchanged, and when planning windows are overlapping, through the use of model MLCLSP in a preceding planning instance enough component production will have been planned to enable production of end item demand forecasted for the periods of the subsequent planning window. Although a completely new production plan may emerge including a new allocation of a predecessor item's available inventory to its successors, with updated outstanding orders and initial inventory for all items, model MLCLSP can be used in a rolling planning mode. The proposed heuristic works with the updated available inventory for each item in each period of the planning window as described in the above example.

Figure 8 **Average Deviation from Lower Bound and Average Computation Time as a Function of the Number of Iterations (Problem Class D)**



4. Computational Results

The procedure was tested using several sets of randomly generated problems. We considered five classes of differently sized problems with up to 100 items/operations, 10 resource types and 16 time periods.

As Trigeiro et al. (1989) point out, in order to have a solution lotsizing problems with heavy utilization must have a very special demand structure, with excess capacity in the first few periods. This is because lotsizing moves production to earlier periods of the planning window, and high mean utilization with excess capacity in earlier periods implies that demand has an increasing trend. In order to achieve a constant average capacity utilization over all periods of the planning horizon for all problems the following procedure for generating end item demand series was applied. For each end item, a mean demand per period was fixed. Then based on this mean demand and a desired coefficient of variation truncated normally distributed observations of demands per period was generated. Available capacity per period was computed by dividing the mean demand by the target capacity utilization. Feasibility of the resulting problem instances with respect to the capacity constraints was maintained by exchanging period demands such that the cumulated capacity available was not less than the cumulated demands. This procedure maintains the desired coefficient of variation but introduces a trend into the demand series.

As lead times (and initial inventory and outstanding orders) would have introduced a further dimension into the generation of problem instances, in all problems lead times and initial inventories and outstanding orders have been neglected. This is in coincidence with the majority of the literature on capacitated lotsizing.

In problems with setup times the effective utilizations of the resources depend on the number of setups, and without ex ante knowledge of the solution it is not possible to generate a problem instance with an exactly preset utilization. For problem classes with setup times we proceeded as follows. Based on the target utilizations (50%, 70%, 90%) the capacities required under a lot-for-lot production policy were computed and used to generate the problem instances. For problem class B it turned out that the resulting mean effective utilizations were very close to the target utilizations. For problem class D, however, for many problem instances the effective utilizations were considerably less than the target utilizations. Therefore we gradually reduced the available capacity in several trial-and-error steps until the mean effective utilizations were close to the target utilizations.

For a discussion of the difficulties arising in the generation of problem instances, including setup times, see Trigeiro et al. (1989).

Table 8 **Average Deviations from the Tempelmeier/Helber Solution (Problem Class E)**

Utilization Profile	1 Iteration	50 Iterations	TBO Profile	1 Iteration	50 Iterations	CV	1 Iteration	50 Iterations
90	−0.81	−2.90	1	−0.19	−0.47	0.1	−2.14	−5.92
70	−5.09	−7.48	3	−7.85	−11.09	0.5	−3.91	−5.99
50	−4.47	−6.56	5	−5.20	−9.25	0.9	−5.03	−6.92
90/70/50	−3.40	−6.43	1/3/5	−5.43	−8.58			
50/70/90	−4.69	−8.02	5/3/1	−0.18	−1.99			
mean	−3.69	−6.28	mean	−3.69	−6.28	mean	−3.69	−6.28

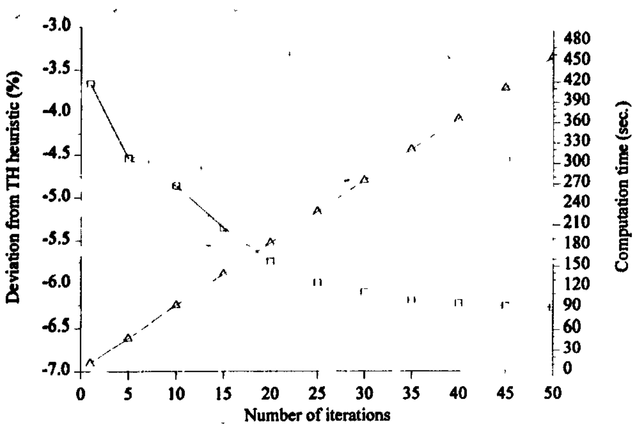
Table 9 Average Deviations of the Heuristic Solutions from the Lower Bound (Problem Class E)

		Utilization Profile					
TBO Profile	CV	90	70	50	90/70/50	50/70/90	mean
1	0.1	0.01%	0.00%	0.00%	0.01%	0.00%	0.01%
	0.5	23.80%	1.62%	0.10%	21.84%	3.83%	10.24%
	0.9	29.10%	6.50%	0.17%	26.57%	7.63%	13.99%
	mean	17.64%	2.71%	0.09%	16.14%	3.82%	8.08%
3	0.1	23.72%	13.39%	9.29%	14.94%	18.94%	16.06%
	0.5	25.61%	12.88%	6.72%	13.92%	16.90%	15 20%
	0.9	25.67%	10.57%	7.00%	12.07%	14.86%	14.03%
	mean	25.00%	12.28%	7.67%	13.64%	16.90%	15.10%
5	0.1	31.04%	22.56%	13.88%	22.54%	32.92%	24.59%
	0.5	34.97%	24.09%	12.12%	20.03%	29.88%	24.22%
	0.9	36.80%	20.68%	9.66%	18.92%	26.00%	22.41%
	mean	34.27%	22.44%	11.89%	20.50%	29.60%	23.74%
1/3/5	0.1	24.56%	14.50%	9.81%	12.05%	21.94%	16.57%
	0.5	27.93%	14.23%	6.60%	11.82%	21.34%	16.38%
	0.9	29.00%	15.13%	7.21%	12.91%	20.07%	16.86%
	mean	27.16%	14.62%	7.87%	12.26%	21.11%	16.61%
5/3/1	0.1	33.86%	22.20%	15.22%	23.35%	27.93%	24.51%
	0.5	34.21%	22.36%	21.97%	24.02%	24.68%	25.45%
	0.9	36.67%	22.84%	21.25%	25.73%	23.02%	25.90%
	mean	34.91%	22.47%	19.48%	24.37%	25.21%	25.29%
		overall mean (150 problem instances)					17.76%

For all problems an "outer" iteration limit of 50 and an "inner" iteration limit of 30 was used.

Problem Class A ($K = 10$; $T = 4$; $J = 3$; No Setup Times):

Figure 9 Average Deviation from the Tempelmeier/Helber Solution and Average Computation Time as a Function of the Number of Iterations (Problem Class E)



Problem class A consists of 300 small problems with 10 items, 4 periods, and 3 resources. The individual problems were generated combining:

Four product/operation structures (general and assembly, with cyclic and noncyclic relations of resources in the product/operation structure);

Three end item demand structures (with varying coefficients of variation CV = {0.1, 0.4, 0.7});

Five combinations of time-between-orders (TBO profiles; mean length of a production cycle between one and four periods); and

Figure 10 General and Assembly Product/Operation Structure for Problem Class A

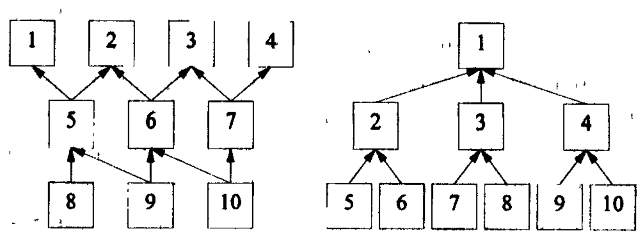


Table 10 Assignment of Products to Resources for Problem Class A

	General Product Structure		Assembly Product Structure	
Resource	A1—noncyclic	A2—cyclic	A3—noncyclic	A4—cyclic
A	1···4	1, 2, 6	1	1, 2
B	5···7	3, 4, 7	2···4	3 ··· 5, 7
C	8···10	5, 8·· 10	5···10	6, 8···10

Five combinations of capacity utilization (utilization profiles; $\rho = \{90\%, 70\%, 50\%\}$).

A detailed description of this problem class is given in the appendix. For each coefficient of variation, five end item demand series were randomly generated. These demand series were used for all combinations of TBO profiles and utilization profiles. Therefore, for a given coefficient of variation, the differences in the average solution quality observed are due to the differences of the TBO profile or the utilization profile. In total 1500 problem instances were considered, for which Tempelmeier and Helber (1994) have computed the optimal solutions using the LINDO-software. All computations were performed on a 80486/33 Mhz PC.

In Table 3 the results are broken down according to utilization profile, TBO profile and coefficient of variation. The numbers divided by slashes denote the mean utilizations (in percent) or TBO values for the upper, middle and lower stages of the product/operation structure.

Figure 6 shows the development of the solution quality as a function of the number of the outer iterations. Observe that after one iteration the mean deviation from optimality is 3.54%.

PROBLEM CLASS B ($K = 10$; $T = 4$; $J = 3$; SETUP TIMES)

Problem class B is generated combining the problems of class A with two setup time profiles.

Setup times varied from 5 to 15 time units, with one time unit being the production time for one unit of an item. In the first setup time profile high setup times are assigned to products/operations located near the end product level, whereas in the second setup time profile the lower level products/operations are assigned higher setup times. Due to the limited computation time available for the computation of the exact solutions we generated one demand time series for each of the resulting 300 problems. Thus a total of 600 problem instances were solved to optimality as well as with the heuristic procedure. The results are presented in Table 4.

With respect to the computation time required and the development of the solution quality, no major differences from problem class A were observed.

In order to gain further insights we changed the target utilizations from 50%, 70%, and 90% to 59%, 79%, and 99%, respectively, which resulted in mean effective utilizations of {98%, 78%, 57%, 98%/78%/57%, 58%/78%/98%}. For three out of 600 problem instances the heuristic procedure did not find a feasible solution, although one existed. With a further increase of the target utilizations to 60%, 80%, and 100% (effective utilizations {99%, 79%, 58%, 99%/79%/58%, 59%/79%/99%}) for 11 out of 600 problem instances no feasible solution was found, although one existed. For these small problems we were able to prove the existence of a solution with LINDO.

PROBLEM CLASS C ($K = 40$; $T = 16$; $J = 6$; NO SETUP TIMES)

Problem class C consists of 300 problems including 40 items, 16 periods, and 6 resources. The individual problems were generated combining:

Table 11 Utilization Profiles for Problem Class A

Utilization Profile	90%	70%	50%
1	A, B, C	–	–
2	–	A, B, C	–
3	–	–	A, B, C
4	A	B	C
5	C	B	A

Table 12 TBO Profiles for Problem Class A

	General Product Structure			Assembly Product Structure		
TBO Profile	TBO = 1	TBO = 2	TBO = 4	TBO = 1	TBO = 2	TBO = 4
1	1···10	-	-	1···10	-	-
2	-	1···10	-	-	1···10	-
3	-	-	1···10	-	-	1···10
4	1···4	5···7	8···10	1	2···4	5···10
5	8···10	5···7	1···4	5···10	2···4	1

Four product/operation structures (general and assembly, with cyclic and noncyclic resource graphs);

Three end item demand structures (with varying coefficients of variation CV = {0.1, 0.5, 0.9});

Five combinations of time-between-orders (TBO profiles; mean length of a production cycle between one and five periods); and

Five combinations of capacity utilization (utilization profiles; ρ = {90%, 70%, 50%}).

For each coefficient of variation two end item demand series were generated which resulted in 600 problem instances. A detailed description of this problem class is given in the appendix. It is very difficult to solve problems of this size exactly—if possible at all. The computational burden of the exact solution excludes the evaluation of the results of the heuristic with respect to optimality. To provide insights concerning the proposed heuristic we compare the results with the results found by the procedure of Tempelmeier and Helber as well as to lower bounds. Table 5 shows the comparison with the solution of the Tempelmeier and Helber procedure.

The development of the solution quality and the computation times are shown in Figure 7.

In Table 6 the results are compared to the lower bounds. The deviations of the heuristic solutions from the lower bounds *in percent of the upper bound* solution, (UB − LB)/UB, are shown.

Problem Class D ($K = 40$; $T = 16$; $J = 6$; Setup Times)

Problem class D is generated based on problem class C by adding four setup time profiles (two for the general and two for the assembly product/operation structure) and considering only one randomly generated end item demand series.

Due to the presence of setup times the Tempelmeier/Helber heuristic is not applicable; therefore, we can base our judgment on the solution quality solely on lower bounds. Table 7 shows the average deviations of the solutions from the lower bounds in percent of the upper bounds.

For test purposes we also increased the target utilizations above 90%. However, in this case for many problem instances, the heuristic did not find a feasible solution. Unfortunately, due to the complexity of the

Table 13 Setup Time Profiles Problem Class B

	Setup Time		
Setup Time Profile	5	10	15
1	7···10	1, 2, 5, 6	3, 4
2	3, 4	1, 2, 5, 6	7···10

Figure 11 General Product/Operation Structure for Problem Class C

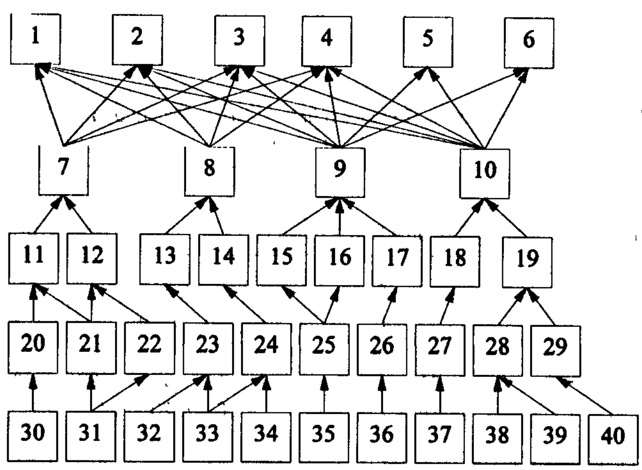
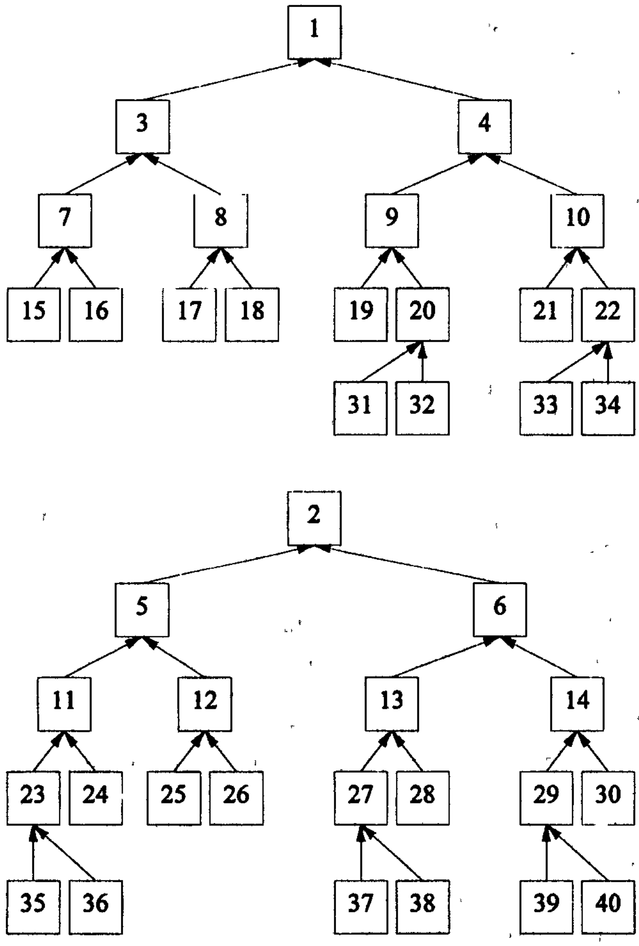


Figure 12 **Assembly Product/Operation Structure for Problem Class C**



feasibility problem (640 integer variables), we were not able to prove with LINDO that a feasible solution existed for any of these problem instances.

Table 14 **Assignment of Products in the General Structure to Resources for Problem Class C**

Resource	C1—noncyclic Resource Graph	C2—cyclic Resource Graph
A	1···3	1···3, 5
B	4···6	4, 6, 18, 19
C	7···10	7···10
D	11···19	11···15, 25···29, 39, 40
E	20···29	16, 17, 20···24, 31
F	30···40	30, 32···38

Table 15 **Assignment of Products in the Assembly Structure to Resources for Problem Class C**

Resource	C3–noncyclic Resource Graph	C4–cyclic Resource Graph
A	1, 2	1···3
B	3···6	4, 6
C	7···14	5, 7···16
D	15···22	17···20, 27, 28, 31, 32, 37, 38
E	23···30	21···23, 33···36
F	31···40	24···26, 29···40

The development of the solution quality and the computation times as a function of the number of iterations are shown in Figure 8.

Problem Class E ($K = 100$; $T = 16$; $J = 10$; No Setup Times)

Problem class E comprises 150 problems with a general product/operation structure including 100 items spread over 10 levels, 16 periods, and 10 resources. The individual problems were generated combining:

Two product/operation structures (general, with cyclic and noncyclic relations between the resources in the product/operation structure);

Three end item demand structures (with varying coefficients of variation CV = {0.1, 0.5, 0.9});

Five combinations of time-between-orders (TBO profiles; mean length of a production cycle between one and five periods); and

Five combinations of capacity utilization (utilization profiles; ρ = {90%, 70%, 50%}).

Further details are presented in the appendix. As no setup times are considered, the results can be compared to the solutions found by the Tempelmeier and Helber

Table 16 **Utilization Profiles for Problem Class C**

Utilization Profile	90%	70%	50%
1	A···F	–	–
2	–	A···F	–
3	–	–	A···F
4	A, B	C, D	E, F
5	E, F	C, D	A, B

Table 17 TBO Profiles for Problem Class C

TBO Profile	TBO = 1	TBO = 3	TBO = 5
1	1···40	–	–
2	–	1···40	–
3	–	–	1···40
4	1···6	7···19	20···40
5	20···40	7···19	1···6

Table 18 Setup Time Profiles Problem Class D

Setup Time Profile (General Product Structure)	Setup Time			
	20	30	40	50
1	31···40	21···30	11···20	1···10
2	1···10	11···20	21···30	31···40
Setup Time Profile (Assembly Product Structure)	**Setup Time**			
	10	15	20	25
3	31···40	21···30	11···20	1···10
4	1···10	11···20	21···30	31···40

heuristic, which took 109 seconds of computation time on the average to solve one problem instance (see Table 8).

Observe that the quality of the solutions compared to the lower bound is in the same order of magnitude as with the much smaller problem class C. Figure 9 shows the development of the solution quality compared to the procedure of Tempelmeier and Helber and the computation time.

The gap between UB and LB is quite large. This makes it difficult to judge the quality of the heuristic with respect to large problems. The question arises where the optimum might be located within the intervall [LB, UB]. To make this point clearer, for all problems for which we know the optimal solution (problem classes A and B and some larger problems with eight periods solved by Tempelmeier and Helber) we compared the ratio of the difference between optimum and LB to the difference between UB and LB, i.e., we computed (Optimum − LB)/(UB − LB). Taking all small problems together, (Optimum − LB) is responsible for more than 80% of the difference (UB − LB). This suggests that the gap between the best solution found and the optimal solution for the large problems may be as much as 80% smaller than the numbers shown in the tables.

It seems that the heuristic particularly performs very well on problems with short TBOs. Short TBOs will result when setup costs are low. In industrial practice setup costs often are set high to account for opportunity costs of capacity lost due to frequent setups. When model MLCLSP is used for lot sizing, then it must be kept in mind that under direct consideration of capacities and setup times the setup costs will loose their character of opportunity

Figure 13 General Product/Operation Structure for Problem Class E

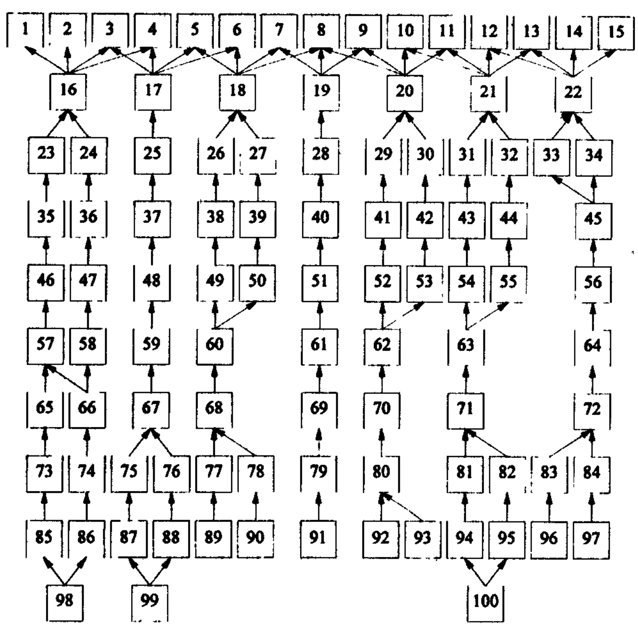


Table 19 Assignment of Products to Resources for Problem Class E

Resource	E1—noncyclic Resource Graph	E2—cyclic Resource Graph
A	1···15	1···7, 49, 50
B	16···22	8···15, 23, 24, 35, 36, 51···54
C	23···34	16···22
D	35···45	25···34, 46···48
E	46···56	37···45, 55, 56, 69, 70, 79, 80
F	57···64	57···64
G	65···72	65···68, 73···78
H	73···84	71, 72, 81···84
I	85···97	85···93, 98, 99
J	98···100	94···97, 100

Table 20 Utilization Profiles for Problem Class E

Utilization Profile	90%	70%	50%
1	A···J	–	–
2	–	A···J	–
3	–	–	A···J
4	A···C	D···G	H···J
5	H···J	D···G	A···C

Table 22 TBO Profiles for Problem Class E

TBO Profile	TBO = 1	TBO = 3	TBO = 5
1	1···100	–	–
2	–	1···100	–
3	–	–	1···100
4	1···33	34···66	67···100
5	67···100	34···66	1···33

costs and will dramatically reduce. Only a small portion will remain comprising the cost component (cleaning materials etc.) directly relevant for lotsizing decisions. In many cases setup costs will be zero. Therefore it can be expected that real-life problems will have setup costs and TBOs that result in "easy" problems for which good heuristic solutions can be found.

5. Conclusion

In this paper we have proposed a heuristic procedure for the dynamic multi-level multi-item capacitated lotsizing problem for general product/operation structures with multiple resources. We have demonstrated how the model can be used within an rolling scheduling framework.

The possibility to take account of setup times relieves the planner from the often nearly unsolvable problem to determine setup costs in the sense of opportunity costs of the capacity lost due to setups. Taking finite capacity directly into consideration during the lotsizing phase will improve the performance of standard MRP systems, as lead times become more predictable. The heuristic procedure proposed could be used as part of a distributed resource-oriented MRP environment, where a workstation is dedicated to a shop for which it performs all lotsizing and scheduling tasks. Research on this issue is on the way.[1]

Table 21 Average Demands for End Products (Problem Class E)

Product	Average Demand	Product	Average Demand
1	100	9	70
2	40	10	120
3	120	11	100
4	100	12	40
5	130	13	40
6	50	14	40
7	60	15	60
8	120		

Appendix

Description of the Problem Classes

Problem class A ($K = 10, T = 4, J = 3$; no setup times)

For the general product structure average demands per period for the end products were set to 70, 30, 50, and 100, respectively. The average demand per period for the end product in the assembly structure was set to 100. Different assignments of products to the resources result in cyclic or non cyclic relations between the resources in the product/operation structure (Table 10).

Problem class B ($K = 10, T = 4, J = 3$; setup times)

Problem class B is based on the combination of the problems of class A with two setup time profiles (Table 13).

Problem class C ($K = 40, T = 16, J = 6$; no setup times)

The average demands of the six end products in the general product structure were set to 40, 20, 30, 60, 20, and 30. For the two end products of the assembly type product structure the average demands were set to 20 and 25.

Problem class D ($K = 40, T = 16, J = 6$; setup times)

Problem class E ($K = 100, T = 16, J = 10$; no setup times)

References

Bahl, H. C., L. P. Ritzman, and J. N. D. Gupta, "Determining Lot Sizes and Resource Requirements: A Review," *Oper. Res.*, 35 (1987), 329–345.
Billington, P. J., "Multi-level Lot-sizing with a Bottleneck Work Center," Ph.D. Dissertation, Cornell University, Ithaca, NY, 1983.
——, J. O McClain, and L. J. Thomas, "Heuristics for Multilevel Lotsizing with a Bottleneck," *Management Sci.*, 32 (1986), 989–1006.

[1] The problem data are available from the first author upon request. The authors are grateful for the helpful comments of the anonymous reviewers.

Blackburn, J. D. and R. A. Millen, "Improved Heuristics for Multi-Stage Requirements Planning Systems," *Management Sci.*, 28 (1982), 44–56.

Chen, W.-H. and J.-M. Thizy, "Analysis of Relaxations for the Multi-item Capacitated Lot-sizing Problem," *Annals of Oper. Res.*, 26 (1990), 29–72.

Crowder, H., "Computational Improvements for Subgradient Optimization," *Symposia Mathematica*, 19 (1976), 357–372.

Diaby, M., H. C. Bahl, M. H. Karwan, and S. Zionts, "A Lagrangean Relaxation Approach for Very-large-scale Capacitated Lot-sizing," *Management Sci.*, 38 (1992), 1329–1340.

Dixon, P. S. and E. A. Silver, "A Heuristic Solution Procedure for Multi-item, Single-level, Limited Capacity Lot-sizing Problem," *J. Oper. Management*, 2 (1981), 22–39.

Federgruen, A. and M. Tzur, "A Simple Forward Algorithm to Solve General Dynamic Lot Sizing Models with n Periods in $O(n \log n)$ or $O(n)$ Time," *Management Sci.*, 37 (1991), 909–925.

Fisher, M. L., "The Lagrangian Relaxation Method for Solving Integer Programming Problems," *Management Sci.*, 27 (1981), 1–18.

Fleischmann, B., "The Discrete Lot-sizing and Scheduling Problem," *European J. Oper. Res.*, 44 (1990), 337–348.

Gupta, Y. P., Y. K. Keung, and M. C. Gupta, "Comparative Analysis of Lot-sizing Models for Multi-stage Systems: A Simulation Study," *International J. Production Res.*, 30 (1990), 695–716.

Heinrich, C. E. and C. Schneeweiß, "Multi-Stage Lot-Sizing for General Production Systems," in S. Axsäter, C. Schneeweiß, and E. A. Silver (Eds.), *Multi-Stage Production Planning and Inventory Control*. Springer, Berlin, 1986.

Helber, S., Kapazitätsorientierte Losgrößenplanung in PPS-Systemen, Dissertation, M und P Verlag, Stuttgart (in German), 1994.

——, Lot Sizing in Capacitated Production Planning and Control Systems, *Oper. Res. Spektrum*, 17 (1995), 5–18.

Held, M., P. Wolfe, and H. P. Crowder, "Validation of Subgradient Optimization," *Math. Programming*, 6 (1974), 62–88.

Kuik, R., M. Salomon, and L. N. Van Wassenhove, "Batching Decisions: Structure and Models, *European J. Oper. Res.*, 75 (1974), 243–263.

Lozano, S., J. Larraneta, and L. Onieva, "Primal-dual Approach to the Single Level Capacitated Lot-Sizing Problem," *European J. Oper. Res.*, 51 (1991), 354–366.

Maes, J., "Capacitated Lotsizing Techniques in Manufacturing Resource Planning," Ph.D. Dissertation, Katholieke Universiteit Leuven, Fakulteit der Toegepaste Wetenschappen, Leuven, 1987.

—— and L. N. Van Wassenhove, "A Simple Heuristic for the Multi-item Single Level Capacitated Lotsizing Problem," *OR Letters*, 4 (1986), 265–273.

—— and ——, "Capacitated Dynamic Lotsizing Heuristics for Serial Systems," *International J. Production Res.*, 29 (1991), 1235–1249.

——, J. O. McClain, and L. N. Van Wassenhove, "Multilevel Capacitated Lotsizing Complexity and LP-Based Heuristics," *European J. Oper. Res.*, 53 (1991), 131–148.

Nemhauser, G. L. and L. A. Wolsey, *Integer and Combinatorial Optimization*. Wiley, New York, 1988.

Pochet, Y. and L. A. Wolsey, "Solving Multi-item Lot-sizing Problems using Strong Cutting Planes," *Management Sci.*, 37 (1991), 53–67.

Salomon, M., *Deterministic lotsizing models for production planning*, Springer, Berlin, 1991.

Shapiro, J. F., "A Survey of Lagrangean Techniques for Discrete Optimization," *Ann. Discrete Math.*, 5 (1979), 113–138.

Shor, N. Z., "The Rate of Convergence of the Generalized Gradient Descent Method," *Cybernetics*, 4, 3 (1968), 79–80.

Tempelmeier, H. and S. Helber, "A Heuristic for Dynamic Multi-item Multi-level Capacitated Lotsizing for General Product Structures," *European J. Oper. Res.*, 75 (1994), 296–311.

Thizy, J.-M., "Analysis of Lagrangian Decomposition for the Multi-item Capacitated Lot-sizing Problem," *INFOR*, 29 (1991), 271–283.

—— and L. N. Van Wassenhove, "Lagrangean Relaxation for Multi-Item Capacitated Lot-sizing Problem: A Heuristic Implementation," *IIE Transactions*, 17 (1985), 308–313.

Trigeiro, W. W., "A Dual-cost Heuristic for the Capacitated Lot Sizing Problem," *IIE Transactions*, 19 (1987), 67–72.

——, L. J. Thomas, and J. O. McClain, "Capacitated Lot Sizing with Setup Times," *Management Sci*, 35 (1989), 353–366.

Wagelmans, A., S. Van Hoesel, and A. Kolen, "Economic Lot Sizing: An $O(n \log n)$ Algorithm that Runs in Linear Time in the Wagner-Whitin Case," *Oper. Res.*, 40 (1992), 145–155.

Accepted by Luk Van Wassenhove; received December 2, 1993. This paper has been with the authors 7 months for 2 revisions.