OR Spectrum (2009) 31:385–404
DOI 10.1007/s00291-008-0130-y

REGULAR ARTICLE

A heuristic for the dynamic multi-level capacitated lotsizing problem with linked lotsizes for general product structures

Horst Tempelmeier · Lisbeth Buschkühl

Published online: 29 February 2008

Abstract In this paper, a new model formulation for the dynamic multi-level capacitated lotsizing problem with linked lotsizes is introduced. Linked lotsizes means that the model formulation correctly accounts for setup carryovers between adjacent periods if production of a product is continued in the next period. This model formulation is a good compromise between the big-bucket and small-bucket model formulation in that it inherits the stability of a big-bucket model and at least partially includes the precise description of setup operations provided by a small-bucket model. A Lagrangean heuristic is developed and tested in a numerical experiment with a set of invented data and a data set taken from industry. The solutions found show a good quality.

Keywords Lotsizing · Multi-level · Setup carry-over

1 Introduction

Many real-life production processes can only start after a setup of the resources with associated setup time and/or setup costs has been completed. Depending on the layout type of the production system (job shop, flow line, flexible manufacturing system, etc.) a variety of different lotsizing problems may occur. Although in the standard MRP approach which is the basis for most software systems lotsizing is only treated in its simplest form, i.e. isolated for each item and without consideration of capacity constraints, in the literature a large number of model variations and solution approaches have been proposed. The dynamic capacitated lotsizing models available differ mainly in the extent to which lotsizing and sequencing decisions are considered

H. Tempelmeier (✉) · L. Buschkühl
Seminar für Supply Chain Management und Produktion,
Universität zu Köln, Cologne, Germany
e-mail: tempelmeier@wiso.uni-koeln.de

simultaneously. Big-bucket model formulations such as the (single-level) capacitated lotsizing problem (CLSP) and its multi-level extension, the multi-level capacitated lotsizing problem (MLCLSP), determine production quantities and periods only, without consideration of the actual production sequence of the orders within a time period. This type of modelling has the virtue that it allows a flexible re-sequencing of orders within a period, at the cost, however, that a detailed production plan must be generated in a sequent planning step. On the other hand, a number of model variations completely integrate lotsizing and sequencing decisions. These small-bucket models are known as the discrete lotsizing and scheduling problem (DLSP), the continuous setup lotsizing problem (CSLP) and the proportional lotsizing and scheduling problem (PLSP), among others. For recent detailed reviews on these models see e.g. Meyr (1999), Staggemeier and Clark (2001) and Sürie (2005).

In order to ensure that an MLCLSP solution can be transformed into a feasible production schedule, a minimum planned lead time of one period must be introduced for each component product. In a multi-level bill-of-material (BOM) structure the cumulated flow time from the beginning of the processing of the raw material to the completion of the finished product is then equal to the number of levels in the BOM structure. As for a period length of 1 week with a ten-level BOM structure this would result in a minimum flow time of at least ten weeks, the cumulated flow time can only be shortened by a reduction of the period length to, say, one day. However, if the time buckets are too small, then only a small number of products will be produced within a single period and in this case it will often happen that production in period t is continued in period $t+1$ (and possibly periods $t+2,\ldots$) without an additional setup. The standard big-bucket MLCLSP formulation counts this as a second setup, and therefore with short period lengths provides only a rough estimate of the real number of setups.

In order to avoid this problem, one could apply a simultaneous lotsizing and sequencing model. The main difference of such a small-bucket model compared to big-bucket lotsizing models is that it correctly counts the number of setups. However, this modelling advantage is achieved at the cost of a significant increase of complexity and a high sensitivity to the change of the planning data which introduces nervousness into the planning process. Setups are reduced, because the setup state can be preserved across periods, whenever the same item is produced at the end of one period and at the beginning of the next. This reduces setup costs as well as setup time and sometimes is the only way to find a feasible solution.

From the perspective of a practical application, a good compromise between planning stability and precision of setup modelling is the capacitated lotsizing problem with linked lot sizes (CLSP-L) (see Haase 1994, 1998). This is a big-bucket model and therefore allows the production of any number of products within a period, but it incorporates partial sequencing of the production orders in the sense that the first and the last product produced in a period are part of a feasible solution of the model, thereby preserving the setup state across periods. Besides being less complex than a small-bucket model formulation, the CLSP-L has the virtue that it provides significant planning flexibility to the planner, who as a reaction of unforeseen events can change the sequence of those products which are not produced across the period borders.

Parallel to the numerous different modelling approaches a great variety of solution procedures for lotsizing problems have been developed. However, despite the large progress that has been achieved, at present exact solution methods are successful for relatively small problems only. Thus, from a practical point of view, when hundreds of products and dozens of periods are to be considered, heuristic solution methods are required. Recent reviews of available modelling and solution approaches are provided by Drexl and Kimms (1997), Salomon (1991), Staggemeier and Clark (2001), Karimi et al. (2003) and Jans and Degraeve (2004).

In this paper, a new model formulation of the multi-level capacitated lotsizing problem with linked lotsizes (MLCLSP-L) is presented, which is the multi-level extension of the CLSP-L. We propose a Lagrangean heuristic to solve this problem. The procedure is closely related to the heuristic of Tempelmeier and Derstroff (1996) (T&D) which was designed for the MLCLSP. We now explicitly handle setup carryovers and we consider a deterministic planned lead time of exactly one period. In capacitated dynamic lotsizing models, lead times greater than one period (to account for congestion at the resources, as applied in the MRP approach) do not make sense, as any feasible solution guarantees that no congestion-related waiting times may occur. Additional time such as transportation time can be accounted for by including the transportation operation into the multi-level lotsizing model. We test the heuristic with respect to data set B of Tempelmeier and Derstroff (1996), to a new data set of 1920 problem instances and to a data set taken from industry with two production stages and a total of 77 end items, 20 components and 70 periods.

In the remainder, we first present a model formulation of the MLCLSP-L. Thereby we focus on the situation with a single setup carryover. Next, we describe a solution procedure. Finally, the results of a computational test are presented.

2 Model formulation

We consider a generally structured multi-level BOM structure with several end products, each with dynamic external period demands over a finite planning horizon. Each item is produced on a single resource with finite period capacity. A setup may cause setup costs as well as setup time. If the production of the last item produced at the end of period t is continued at the beginning of the next period $t+1$, no additional setup is required. The problem is to find the cost-minimal production plan.

In particular, the following assumptions are in effect:

- The planning horizon is divided into T periods (usually shifts or days).
- There are M resources with period-specific capacities.
- K items with dynamic external period demands are arranged in a general product/process structure with a unique assignment of each item to a single resource. The production of a product requires variable production time and fixed setup time.
- Setups are assumed to be sequence-independent.
- A setup is carried over from one period to the next at most once.
- Holding costs per unit and period are applied to the inventory at the end of a period.
- Backorders are not allowed.
- The planned lead time for each product is one period.

– The objective is to minimize the sum of holding costs, setup costs, and variable production costs.

We use the following notation.

Indices and sets:

k	index of items, $k = 1, 2, \ldots, K$
$\mathcal{K}_m$	set of items k that are produced on resource m
m	resource index, $m = 1, 2, \ldots, M$
$\mathcal{P}_k$	set of predecessors of item k
$\mathcal{S}_k$	set of immediate successors of item k
t	period index, $t = 1, 2, \ldots T$.

Data:

a_{kj}	Gozinto factor (quantity of item k which is directly required to produce one unit of item j)
b_t^m	capacity of resource m in period t (in time units)
d_{kt}	external demand of item k in period t
D_{kt}	total demand of item k in period t; $D_{kt} = d_{kt} + \sum_{j \in \mathcal{S}_k} a_{kj} \cdot D_{jt}$; $D_{k,T+1=0}$
e_k	echelon holding cost for item k
h_k	full holding cost for item k
Ω	a very large number
p_{kt}	variable production cost for item k in period t
s_k	setup cost for item k
tb_k	variable production time per unit of item k
tr_k	setup time for item k (on its associated resource)
$\hat{y}_k$	initial inventory of item k at the beginning of the planning horizon.

Decision variables:

γ_{kt}	binary setup (action) variable for item k in period t
ω_{kt}	binary setup state variable for item k in the beginning of period t
q_{kt}	production quantity of item k in period t; $q_{k,T+1} = 0$; note that a production lot may be composed of the production quantities of two consecutive periods (i. e. $q_{kt} + q_{k,t+1}$).

The model formulation is in the line of the MLCLSP formulation of Billington et al. (1986) and Tempelmeier and Derstroff (1996) and reads as follows:

Model MLCLSP-L

$$Z = \sum_{k=1}^{K} \left\{ \sum_{t=1}^{T} (s_k \cdot \gamma_{kt} + [p_{kt} + e_k \cdot (T - t + 1)] \cdot q_{kt}) + (h_k - e_k) \cdot q_{k1} \right\} - \mathrm{C} \tag{1}$$

subject to

$$\sum_{\tau=1}^{t} (q_{k\tau} - d_{k\tau}) + \hat{y}_k \geq \sum_{j \in \mathcal{S}_k} \sum_{\tau=0}^{t} a_{kj} \cdot q_{j,\tau+1} \quad k = 1, 2, \ldots, K; \quad t = 0, 1, \ldots, T \tag{2}$$

$$\sum_{k\in\mathcal{K}_m} (tb_k \cdot q_{kt} + tr_k \cdot \gamma_{kt}) \le b_t^m \qquad m = 1, 2, \ldots, M; \ \ t = 1, 2, \ldots, T \tag{3}$$

$$q_{kt} \le (\gamma_{kt} + \omega_{kt}) \cdot \Omega \qquad k = 1, 2, \ldots, K; \ \ t = 1, 2, \ldots, T \tag{4}$$

$$\sum_{k\in\mathcal{K}_m} \omega_{kt} \le 1 \qquad m = 1, 2, \ldots, M; \ \ t = 1, 2, \ldots, T \tag{5}$$

$$\omega_{kt} \le \gamma_{k,t-1} \qquad k = 1, 2, \ldots, K; \ \ t = 2, 3, \ldots, T \tag{6}$$

$$q_{kt} \ge 0 \qquad k = 1, 2, \ldots, K; \ \ t = 1, 2, \ldots, T \tag{7}$$

$$\gamma_{kt}, \omega_{kt} \in \{0, 1\} \qquad k = 1, 2, \ldots, K; \ \ t = 1, 2, \ldots, T \tag{8}$$

with

$$\mathrm{C} = \sum_{k=1}^{K}\sum_{t=1}^{T} h_k \cdot ((T - t + 1) \cdot d_{kt} - \widehat{y}_k) + \sum_{k=1}^{K}\sum_{t=1}^{T} (h_k - e_k) \cdot D_{kt} \tag{9}$$

The objective function (1) minimizes the sum of setup costs, variable production costs and inventory holding costs. Constraints (2) are the inventory balance constraints, which state that for each time interval $[1, t]$ the cumulated production quantity that is left over after subtraction of the external demand (left-hand side) must be sufficient to cover the cumulated derived demands in the interval $[1, t + 1]$ (right-hand side). Note that the planned lead time for derived demands is one period. Constraints (3) are the capacity constraints requesting that for each resource type m and period t the sum of setup and production time for all products produced on that resource must not exceed the available capacity. Constraints (4) ensure that production of item k takes place in period t only if the resource is setup for this item by a setup operation during that period ($\gamma_{kt} = 1$) or if the resource is already in the correct setup state at the beginning of that period ($\omega_{kt} = 1$). Each resource can be only in one setup state at the beginning of each period. This is requested by constraints (5). For a setup state to be carried over to period t, there must have been a corresponding setup operation in period $t - 1$, according to constraints (6). Finally, q_{kt} must be non-negative and γ_{kt} and ω_{kt} are defined as binary variables. The derivation of the constant C is given in the Appendix.

Model MLCLSP-L reduces to the standard MLCLSP (with no setup carryovers), if $\omega_{kt} = 0$ ($k = 1, 2, \ldots, K, t = 1, 2, \ldots, T$). See also Tempelmeier and Derstroff (1996).

3 Solution method

As model MLCLSP-L can be solved with standard mixed-integer programming software only for very small problem sizes with no practical relevance, we propose a heuristic solution approach based on Lagrangean relaxation.

The heuristic is composed of the following steps. First, through the Lagrangean relaxation of the inventory balance constraints (2), the capacity constraints (3) and the setup carryover constraints (5), the multilevel multi-item capacitated dynamic lotsizing problem with setup carryovers is decomposed into multiple *single-item uncapacitated dynamic lotsizing problems with setup carryovers*. These are solved with a dynamic programming algorithm. The optimal solutions are then used to compute an actual lower bound. Next, the violations of the relaxed constraints by the current solution are computed and used to adjust the Lagrangean multipliers applied in the next iteration. Finally, a feasible solution to problem MLCLSP-L is constructed which provides an upper bound to the optimal objective value. These steps are performed iteratively and will be detailed in the sequel.

3.1 Computation of the lower bound

Introducing Lagrangean multipliers **u** (for the inventory balance constraints), **v** (for the capacity constraints) and **w** (for the setup carryover constraints), we obtain the relaxed model MLCLSP-L-LR.

Model MLCLSP-L-LR

$$\begin{aligned}\text{Minimize} L(\mathbf{u},\mathbf{v},\mathbf{w}) = \sum_{m=1}^{M}\sum_{k\in\mathcal{K}_m}\sum_{t=1}^{T}\Bigg[& p_{kt} + e_k\cdot(T-t+1) \\ & + \Bigg(\sum_{j\in\mathcal{V}_k}\sum_{\substack{\tau=t-1\\ \tau>0}}^{T} a_{jk}\cdot u_{j\tau} - \sum_{\tau=t}^{T} u_{k\tau}\Bigg) + v_t^m\cdot tb_k\Bigg]\cdot q_{kt} \\ & + \sum_{m=1}^{M}\sum_{k\in\mathcal{K}_m}\sum_{t=1}^{T}\big((s_k + v_t^m\cdot tr_k)\cdot\gamma_{kt} + w_t^m\cdot\omega_{kt}\big) \\ & + \sum_{m=1}^{M}\sum_{k\in\mathcal{K}_m}(h_k - e_k)\cdot q_{k1} \\ & - \sum_{k=1}^{K}\sum_{t=1}^{T}(h_k - u_{kt})\cdot((T-t+1)\cdot d_{kt} - \widehat{y}_k) \\ & - \sum_{k=1}^{K}\sum_{t=1}^{T}(h_k - e_k)\cdot D_{kt} - \sum_{m=1}^{M}\sum_{t=1}^{T}(v_t^m\cdot b_t^m + w_t^m)\end{aligned} \tag{10}$$

subject to

$$\sum_{\tau=1}^{t}(q_{k\tau} - D_{k\tau} + \hat{y}_k) \ge 0 \quad k=1,2,\ldots,K;\ t=1,2,\ldots,T \tag{11}$$

$$q_{kt} \le (\gamma_{kt} + \omega_{kt})\cdot\Omega \quad k=1,2,\ldots,K;\ t=1,2,\ldots,T \tag{12}$$

$$\omega_{kt} \le \gamma_{k,t-1} \quad k=1,2,\ldots,K;\ t=2,3,\ldots,T \tag{13}$$

$$q_{kt} \ge 0 \quad k=1,2,\ldots,K;\ t=1,2,\ldots,T \tag{14}$$

$$\gamma_{kt},\omega_{kt} \in \{0,1\} \quad k=1,2,\ldots,K;\ t=1,2,\ldots,T. \tag{15}$$

Equations (11) ensure that for each item k the cumulated production until any period t is large enough to fill the cumulated total demand up to that period. Constraints (12) to (15) are identical to constraints (4) and (6) to (8) which have been repeated here for better readability.

Problem MLCLSP-L-LR decomposes into K single-item uncapacitated dynamic lotsizing problems with setup carryovers. As the constant part in the objective function,

$$F = \sum_{k=1}^{K}\sum_{t=1}^{T}(h_k - u_{kt}) \cdot ((T - t + 1) \cdot d_{kt} - \widehat{y}_k) + \sum_{k=1}^{K}\sum_{t=1}^{T}(h_k - e_k) \cdot D_{kt} + \sum_{m=1}^{M}\sum_{t=1}^{T}(v_t^m \cdot b_t^m + w_t^m), \tag{16}$$

does not influence the optimal lotsizing decision, it can be dropped. Thus, for product $k \in \mathcal{K}_m$ that is processed by resource m we obtain model SLULSP-L$_k(\mathbf{u}, \mathbf{v}, \mathbf{w})$.

Model SLULSP-L$_k$(u, v, w)

$$\text{Minimize } Z_k = \sum_{t=1}^{T}(\eta_{kt} \cdot \gamma_{kt} + \theta_{kt} \cdot q_{kt} + w_t^m \cdot \omega_{kt}) \tag{17}$$

subject to

$$\sum_{\tau=1}^{t}(q_{k\tau} - D_{k\tau} + \hat{y}_k) \geq 0 \quad t = 1, \ldots, T \tag{18}$$

$$q_{kt} \leq (\gamma_{kt} + \omega_{kt}) \cdot \Omega \quad t = 1, \ldots, T \tag{19}$$

$$\omega_{kt} \leq \gamma_{k,t-1} \quad t = 2, \ldots, T \tag{20}$$

$$q_{kt} \geq 0 \quad t = 1, \ldots, T \tag{21}$$

$$\gamma_{kt}, \omega_{kt} \in \{0, 1\} \quad t = 1, \ldots, T \tag{22}$$

with

$$\eta_{kt} = s_k + v_t^m \cdot tr_k \quad t = 1, 2, \ldots, T \tag{23}$$

$$\theta_{k1} = p_{k1} + e_k \cdot T + \left(\sum_{j \in \mathcal{V}_k}\sum_{\tau=1}^{T} a_{jk} \cdot u_{j\tau} - \sum_{\tau=1}^{T} u_{k\tau}\right) + v_1^m \cdot tb_k + h_k - e_k \quad t = 1, 2, \ldots, T \tag{24}$$

$$\theta_{kt} = p_{kt} + e_k \cdot (T - t + 1) + \left(\sum_{j \in \mathcal{V}_k}\sum_{\tau=t-1}^{T} a_{jk} \cdot u_{j\tau} - \sum_{\tau=t}^{T} u_{k\tau}\right) + v_t^m \cdot tb_k \quad t = 2, 3, \ldots, T \tag{25}$$

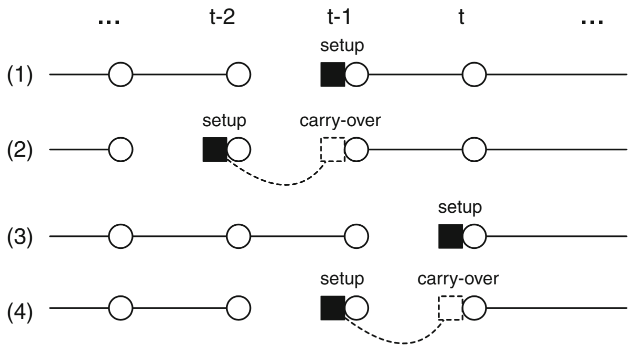

Fig. 1 Different options to cover the demand of period t

As has been pointed out by Sox and Gao (1999), there exists an optimal solution of problem SLULSP-$L_k(\mathbf{u}, \mathbf{v}, \mathbf{w})$ such that each production lot consists of an integer number of period demands. This means that in each period t it is either optimal to cover the complete period demand by the preceding lot or to start a new lot in period t.

The difference between problem SLULSP-$L_k(\mathbf{u}, \mathbf{v}, \mathbf{w})$ and the standard Wagner-Whitin problem is that the setup state may be preserved across two adjacent periods. Thus four alternative options to cover the demand in period t are available:

(1) Add the demand in period t to an existing production lot in a previous period $\tau < t$ which has been installed with a new setup in period τ.
(2) Add the demand in period t to an existing production lot in a previous period $\tau < t$ which has been installed based on a setup carryover from period $\tau - 1$ to period τ.
(3) Perform a setup in period t and produce the demand of period t.
(4) Perform a setup in the preceding period $t - 1$, carry over the setup state and produce the demand of period t.

These alternatives are illustrated in Fig. 1, where the circles denote the period demands, the black rectangles are setups and the white rectangles denote a production lot without a setup.

Problem SLULSP-$L_k(\mathbf{u}, \mathbf{v}, \mathbf{w})$ can be solved with a dynamic programming algorithm similar to the Wagner–Whitin algorithm (Wagner and Whitin 1958; Sox and Gao 1999; Briskorn 2006), whereby the demands of the first periods are adjusted by the initial inventory. Let f_{kt} be the optimal value for the t-period problem of item k. Then the dynamic programming recursion reads as follows:

$$f_{kt} = \min\left\{ \min_{\tau \le t \,\wedge\, \omega_{k\tau}=0} \left\{ \eta_{k\tau} + \theta_{k\tau} \cdot \sum_{s=\tau}^{t} D_{ks} + f_{k,\tau-1} \right\}, \right. \\ \left. \min_{\tau \le t \,\wedge\, \omega_{k\tau}=1} \left\{ \eta_{k,\tau-1} + \theta_{k,\tau-1} \cdot D_{k,\tau-1} + \theta_{k\tau} \cdot \sum_{s=\tau}^{t} D_{ks} + w_{\tau}^{m} + f_{k,\tau-2} \right\} \right\} \tag{26}$$

with $f_{k,-1} = f_{k0} = 0$.

With the solution of the single-item problems, the value of the lower bound is given by Eq. (27).

$$LB = \sum_{m=1}^{M} \sum_{k \in \mathcal{K}_m} Z_k - F \tag{27}$$

3.2 Updating the Lagrangean multipliers

The Lagrangean multipliers are updated similar to Sox and Gao (1999). In any iteration ℓ, the subgradients for **u**, **v** and **w** are exponentially smoothed as follows:

$$\zeta_{kt}^{\ell} = \alpha^{\ell} \cdot \left(\sum_{\tau=1}^{t} \left(d_{k\tau} - q_{k\tau}^{\ell} \right) + \sum_{j \in \mathcal{S}_k} \sum_{\tau=0}^{t} a_{kj} \cdot q_{j,\tau+1}^{\ell} - \widehat{y}_k \right) + (1 - \alpha^{\ell}) \cdot \zeta_{kt}^{\ell-1}$$
$$k = 1, 2, \ldots, K; \quad t = 1, 2, \ldots, T; \quad \ell = 1, 2, \ldots, \ell^{\max} \tag{28}$$

$$\xi_t^{m\ell} = \alpha^{\ell} \cdot \left(\sum_{k \in \mathcal{K}_m} (tb_k \cdot q_{kt}^{\ell} + tr_k \cdot \gamma_{kt}^{\ell}) - b_t^m \right) + (1 - \alpha^{\ell}) \cdot \xi_t^{m,\ell-1}$$
$$m = 1, 2, \ldots, M; \quad t = 1, 2, \ldots, T; \quad \ell = 1, 2, \ldots, \ell^{\max} \tag{29}$$

$$\varsigma_t^{m\ell} = \alpha^{\ell} \cdot \left(\sum_{k \in \mathcal{K}_m} \omega_{kt}^{\ell} - 1 \right) + (1 - \alpha^{\ell}) \cdot \varsigma_t^{m,\ell-1}$$
$$m = 1, 2, \ldots, M; \quad t = 1, 2, \ldots, T; \quad \ell = 1, 2, \ldots, l^{\max} \tag{30}$$

The exponential smoothing parameter is updated with $\alpha^{\ell} = \frac{\alpha^{\ell-1}}{\alpha^{\ell-1} + \alpha^{\text{red}}}$, where $\alpha^1 = 1$ and $\alpha^{\text{red}} = 0.25$. Starting at 0, the multipliers u_{kt}, v_t^m and w_t^m are updated at each iteration using the respective subgradient and the current stepsize λ^{ℓ}:

$$u_{kt}^{\ell} = \max \left\{ 0, u_{kt}^{\ell-1} + \lambda_u^{\ell} \cdot \zeta_{kt}^{\ell} \right\} \quad k = 1, 2, \ldots, K; \; t = 1, 2, \ldots, T \tag{31}$$

$$v_t^{m\ell} = \max \left\{ 0, v_t^{m,\ell-1} + \lambda_v^{\ell} \cdot \xi_t^{m\ell} \right\} \quad m = 1, 2, \ldots, M; \; t = 1, 2, \ldots, T \tag{32}$$

$$w_t^{m\ell} = \max \left\{ 0, w_t^{m,\ell-1} + \lambda_w^{\ell} \cdot \varsigma_t^{m\ell} \right\} \quad m = 1, 2, \ldots, M; \; t = 1, 2, \ldots, T \tag{33}$$

As a result of initial computational tests we use the same stepsize λ_u^{ℓ} for the capacity and inventory balance constraints but a different stepsize λ_w^{ℓ} for the setup carryover constraints. Thus

$$\lambda_u^{\ell} = \lambda_v^{\ell} = \delta^{\ell} \cdot \frac{UB^{\ell-1} - LB^{\ell}}{\sum_{k=1}^{K} \sum_{t=1}^{T} (\bar{\zeta}_{kt}^{\ell})^2 + \sum_{m=1}^{M} \sum_{t=1}^{T} (\bar{\xi}_t^{m\ell})^2} \tag{34}$$

and

$$\lambda_w^\ell = \delta^\ell \cdot \frac{UB^{\ell-1} - LB^{\ell-1}}{\sqrt{\sum_{m=1}^{M} \sum_{t=1}^{T} (\bar{\varsigma}_t^{m\ell})^2}} \tag{35}$$

with $\delta^0 = 2$ and $\delta^\ell = \frac{\delta^{\ell-1}}{2}$, when the value of the lower bound has not been improved for the last four iterations. The symbols $\bar{\zeta}_{kt}$, $\bar{\xi}_t^{m\ell}$ and $\bar{\varsigma}_t^{m\ell}$ denote the amounts of violations of the relaxed constraints which are equivalent to the big round brackets in Eqs. (28), (29) and (30).

The upper bound of the previous iteration, $UB^{\ell-1}$, is used as an estimate for the unknown optimal value Z^*, while $LB^{\ell-1}$ is the current lower bound computed with the Lagrangean multipliers of the previous iteration. The procedure is terminated after 50 iterations, or alternatively, when the Lagrangean multipliers are sufficiently small.

3.3 Computation of the upper bound

In each iteration a feasible solution is generated based on the current solution of the relaxed problem. Feasibility is established by considering the relaxed constraints one by one.

Inventory balance constraints Compliance with the inventory balance constraints can be ensured by processing the items according to their low level code, first computing the period-specific net demand and then solving the corresponding K problems of the type SLULSP-L$_k$.

Setup carryover constraints The next step to reach a feasible solution to the original problem is to ensure that there is not more than one setup carryover per resource and period, i.e. that constraints (5) are met. For each resource m and in each period t, depending on the sum $\sum_{k \in \mathcal{K}_m} \omega_{kt}$ we proceed as follows.

If $\sum_{k \in \mathcal{K}_m} \omega_{kt} < 1$, no setup carryover has been installed in t. Thus, if an item k is produced both in period $t-1$ and in period t and a setup is performed in both periods, then the second setup is replaced by a setup carryover. This saves both setup time and costs. Let $\mathcal{K}_\gamma = \{k : k \in \mathcal{K}_m \wedge \gamma_{k,t-1} + \gamma_{kt} = 2\}$. If $\mathcal{K}_\gamma \neq \emptyset$, then find $k^* = \text{argmax}_{k \in \mathcal{K}_\gamma}\{s_k\}$. Set $\omega_{k^*t} = 1$ and $\gamma_{k^*t} = 0$, in order to maximize the setup cost reduction.

If $\sum_{k \in \mathcal{K}_m} \omega_{kt} = 1$, then exactly one setup carryover has been installed, because ω_{kt} is binary.

If $\sum_{k \in \mathcal{K}_m} \omega_{kt} > 1$, more than one setup carryover has been installed on resource m. In this case all but one setup carryover must be removed. This is done as follows. Let $\mathcal{K}_\omega = \{k : k \in \mathcal{K}_m \wedge \omega_{kt} = 1\}$, then find $k^* = \text{argmax}_{k \in \mathcal{K}_\omega}\{s_k\}$. For all $k \in \mathcal{K}_\omega \backslash k^*$ let $\omega_{kt} = 0$ and $\gamma_{kt} = 1$.

Capacity constraints From the current solution, it remains to remove the capacity violations. To this end we interpret actual excess capacity demands of the lower bound schedule as overtime. Further, we apply a smoothing procedure, which eliminates overtime by iteratively shifting production across periods. Throughout this shifting procedure any violation of the setup carryover constraints and the inventory balance constraints is avoided. The procedure terminates with the first feasible schedule or when a maximum number of iterations has been reached.

The procedure works period by period, i.e. in each period all resources are considered before moving on to the next period. Forward passes starting in period 1 and shifting production quantities into future periods and backward passes starting in period T shifting production into the past are applied alternately. During the first (forward) pass, only cumulative overtime is eliminated. Like Tempelmeier and Derstroff (1996), we found that the additional flexibility for the subsequent backward pass is beneficial. At each move, the currently most beneficial item is selected. Single-item shifting is applied first for all items. However, if this fails to eliminate all overtime, multi-item shifts are applied. This results in four types of shifts: single-item forward, multi-item forward, single-item backward and multi-item backward.

The **single-item forward** pass seeks to shift production quantities incurring overtime in the current period into the future. The shiftable quantity is limited by the respective lotsize, the available capacity in the target period and the restriction, that no backorders may occur in any period. Note that more than the quantity needed to eliminate overtime in the current period is shifted into the future, if possible, in order to reduce holding costs and again to provide additional flexibility for the subsequent backward pass. Capacity in the target period can be exceeded, if this leads to the complete elimination of overtime in the current period. The single-item forward pass ends when either all overtime has been eliminated or there are no shiftable production quantities left.

The **multi-item forward** pass considers (complete) linear subsets of the corresponding item's successors. The maximum shiftable quantity is determined analogously to the single-item shifts for the item furthest away in the bill of material and then exploded to the item in the current period.

The **single-item backward** pass differs from the corresponding forward pass in several aspects. First, backlogging does not have to be avoided for the item considered, but instead for the predecessors, for which derived demand depends on the production period of the selected item. Second, additional overtime in the target period is accepted, if this leads to the elimination of a complete lot and hence to the reduction of setup time and costs. Third, additional cumulated overtime can be incurred.

The **multi-item backward** pass shifts backwards the production quantities of all predecessors along with the selected item, but only into the respectively previous period. This is done by first computing the quantity of the selected item which has to be shifted to eliminate overtime on the corresponding resource in the corresponding period. Then the quantities that have to be shifted to avoid backorders are computed along the bill of material structure.

In the MLCLSP without setup carryovers, there are only two possibilities with respect to the setup state in the target period. Either a setup already exists or it has to be installed, hence incurring setup costs. By contrast, with setup carryovers, four different cases must be considered when shifting production into period t. First, if there is no setup, neither in the current nor in the preceding period ($\gamma_{k,t-1} = \gamma_{kt} = 0$), a new setup is installed in period t. Second, if there is a setup in t ($\gamma_{kt} = 1$), then nothing is changed. Third, if there is a setup for item k in the preceding period ($\gamma_{k,t-1} = 1$)

and there is no setup carryover to period t for a different item $i \neq k$, either there is a setup carryover for item k to t or it can be installed. Finally, if a setup carryover could be installed ($\gamma_{k,t-1} = 1$) but there exists one for a different item i ($\omega_{it} = 1$), the carryover of i is replaced, if $s_k > s_i$ or else a new setup is installed for item k in period t.

In the origin period, a setup can be eliminated in the MLCLSP, if the complete lot is shifted, hence reducing setup costs. Again, there are more options in the MLCLSP-L. First, if there is a setup in t and no carryover to $t+1$, the setup can be eliminated. Second, if there is a setup in t and a setup carryover to period $t+1$ ($\gamma_{kt} = 1, \omega_{k,t+1} = 1$), then the setup cannot be eliminated, but only shifted to the next period $t+1$, if the remaining capacity in $t+1$ suffices. We also search for an alternative setup carryover for a different item i with ($\gamma_{it} = \omega_{i,t+1} = 1$ and $\omega_{i,t+2} = 0$). Then the corresponding setup costs are avoided. Again, this can only be done, if the remaining capacity in $t+1$ after removing the setup for i suffices to cover the setup time of k. Third, if there is no setup in t, but a setup in $t-1$ and a carryover to t, the carryover to t is eliminated. The setup in $t-1$ can be eliminated, if $q_{k,t-1} = 0$ and an alternative carryover is installed as explained above, if possible.

The resources are considered according to the production stage of the corresponding items, such that multi-item shifts are favored. Hence, during forward passes first machines producing items with many successors are selected while during backward passes those producing items with few successors are prioritized. We also tested working resource by resource and in opposite ordering in all possible combinations. However, all of these combinations proved inferior. This corresponds to Tempelmeier and Derstroff (1996) findings for the MLCLSP.

At each move, items are selected according to the corresponding incremental costs, i.e. with the highest cost reduction or the lowest cost increase. The relevant costs are those modified by the Lagrangean multipliers. The aim is to find a feasible solution close to the solution of the relaxed problem. Additionally, cost savings are divided by the total amount of overtime that is eliminated and cost increases multiplied likewise.

3.4 Postoptimization

Following Tempelmeier and Derstroff (1996), we apply several procedures for postoptimization. First, partial or entire lots are shifted into a future period t in order to avoid holding costs and possible setup costs. This is possible, if in period t a production is scheduled ($q_{kt} > 0$) and there is inventory at the end of period $t-1$ ($y_{k,t-1} > 0$) and there is unused capacity available in period t ($b_t^m - \sum_{k \in \mathcal{K}_m} (tr_k \cdot \gamma_{kt} + tb_k \cdot q_{kt}) > 0$).

Second, complete lots are shifted to prior periods as to avoid setup costs, if this is possible and beneficial. Both procedures are applied alternately until no further improvement is achieved.

An additional postoptimization procedure seeks to improve setup decisions. Each resource is considered in a period-by-period approach, starting with period 1. If no setup carryover is installed, we search for a possible setup carryover in the same manner as above. If $\gamma_{k,t-1} = \gamma_{kt} = 1$ and $\omega_{k,t+1} = 0$, a setup carryover can be installed for k in t, hence eliminating the corresponding setup. Out of the possible candidates, item k is selected, which incurs the maximum setup costs (again with respect to the modified costs of the Lagrangean relaxation).

4 Computational study

In order to test the quality of the proposed heuristic, we performed a computational study using a large number of invented problem instances and one data set taken from industrial practice.

Invented data. We first consider a subset of the test instances introduced by Tempelmeier and Derstroff (1996), namely the 600 problem instances of class B with a non-cyclic resource graph, as depicted in Fig. 2. This is the only test set with setup times for which the authors compared theirs results to the exact solutions.

All instances comprise ten items, three resources and six periods. The 600 instances were generated combining:

1. One general and one assembly product structure
2. Three demand structures with varying coefficients of variation (CV)
3. Five setup cost structures resulting in different profiles of average times-between-orders (TBO, average length of a production cycle)
4. Five capacity utilization profiles
5. Two setup time profiles
6. Two resource assignment profiles.

Unlike the original data specification, for each component item we assumed a lead time of one period. Thus, end items can be produced earliest one period after their components have been completed. To achieve feasibility, two production periods without external demand have been added at the beginning of the planning horizon.

The exact solutions of these problems were computed using CPLEX on a Unix workstation using eight UltraSPARC-III-processors in parallel with 0.9 GHz each. The heuristic was run on a standard Pentium, 2.8 GHz, 1 GB RAM. The average computation time per problem instance is about 0.018 seconds. In Table 1 the percentage deviations of the heuristic solution values from the exact values are presented, broken down according to utilization profile, TBO profile, and coefficient of variation of the demand series.

New invented data. To further test our heuristic, we generated a test set of 1,920 problems instances with increased size similar to the way Tempelmeier and Derstroff

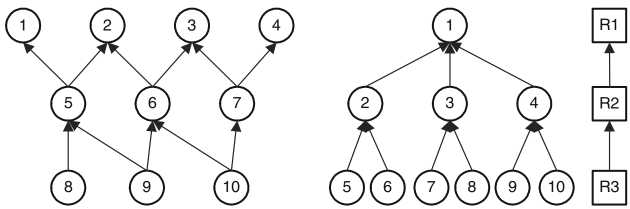

Fig. 2 General and Assembly Product Structure for class B

Table 1 Average Percentage Deviations from Optimality

TBO profile	Utilization profile						
	CV	90(%)	70(%)	50(%)	90/70/50(%)	50/70/90(%)	Mean(%)
1	0.1	0.00	0.00	0.00	0.00	0.00	0.00
	0.4	0.00	0.01	0.16	0.06	0.86	0.22
	0.7	0.00	0.74	2.77	3.45	0.85	1.56
	Mean	0.00	0.25	0.98	1.17	0.57	0.59
2	0.1	1.06	1.03	0.88	0.68	1.01	0.93
	0.4	1.55	1.88	1.23	1.04	2.17	1.58
	0.7	1.29	1.39	1.93	2.06	2.86	1.91
	Mean	1.30	1.43	1.35	1.26	2.01	1.47
3	0.1	4.39	1.15	2.62	4.25	2.25	2.93
	0.4	5.72	3.53	1.83	3.10	5.24	3.88
	0.7	3.00	3.01	3.52	2.64	3.86	3.21
	Mean	4.37	2.56	2.65	3.33	3.78	3.34
4	0.1	2.56	1.11	3.80	4.18	2.03	2.73
	0.4	2.21	1.44	3.03	3.37	1.94	2.40
	0.7	1.32	1.39	1.38	1.87	1.24	1.44
	Mean	2.03	1.31	2.74	3.14	1.73	2.19
5	0.1	3.73	3.43	1.06	1.99	3.47	2.74
	0.4	2.62	2.06	2.63	2.85	2.57	2.55
	0.7	1.20	0.69	3.02	1.89	2.00	1.76
	Mean	2.52	2.06	2.23	2.24	2.68	2.35
				Overall mean (600 problem instances)			1.99

Table 2 Dimensions of the new test problems

Class	♯ Products	♯ Resources	♯ Periods	♯ Instances
1	10	3	4	480
2	10	3	8	480
3	20	6	8	240
4	20	6	16	240
5	40	6	8	240
6	40	6	16	240

(1996) created the test set indicated above. This new test set is divided into six classes with the dimensions given in Table 2.

Furthermore, for each class a general (G) and an assembly (A) product structure is combined with respectively an non-cyclic (N) and a cyclic (C) process structure. Three demand profiles were generated with coefficients of variation of 0.2, 0.5 and 0.8 respectively. Four TBO profiles were used for classes 1 and 2, and two TBO profiles for classes 3 to 6. Furthermore, we considered two setup profiles and three capacity

Table 3 Average deviation per class

Class	$\frac{UB^{heuristic}-LB^{CPLEX}}{LB^{CPLEX}}$ (%)
1	1.39
2	4.75
3	13.76
4	21.90
5	11.29
6	16.61
Overall mean	9.47

Table 4 Average deviation per setup time and capacity profile

Capacity profile	Setup time profile	$\frac{UB^{heuristic}-LB^{CPLEX}}{LB^{CPLEX}}$ (%)
1	1	11.57
1	2	11.51
2	1	10.99
2	2	12.34
3	1	7.41
3	2	7.86
4	1	8.33
4	2	8.70
5	1	7.90
5	2	8.01
Overall mean		9.47

profiles. A detailed description of the test problems are available for download from http://www.scmp.uni-koeln.de/publikationen/ORS2008MLCLSPL.zip.

Although the dimensions of these new problems have been only moderately increased, in many cases we were not able to compute the exact solution within a time limit of one hour on the above-mentioned workstation with eight parallel UltraSPARC-III-processors. The average difference between upper and lower bound relative to the upper bound computed over all problem instances found with CPLEX was 0.1263%. The average deviation of the heuristic is given as the difference between the upper bound found with the heuristic and the lower bound computed with CPLEX over the lower bound computed with CPLEX. The average time required by the heuristic was 0.15 CPU seconds.

For four instances in class 6, no feasible solution was found. In Tables 3 – 6, for the remaining 1,914 instances the results are broken down with respect to setup time and capacity profile, TBO profile, and product and process structure.

From the numerical results we draw the following conclusions. It appears that the heuristic performs better for problem instances with less periods compared to the number of products (see Table 3). In addition, the heuristic works better for problems with higher utilizations than for problems with low utilized resources (see Table 4). Note that capacity profiles 1 and 2 represent a target utilization of 50 and 70%, while

Table 5 Average deviation per TBO profile

TBO profile	$\frac{UB^{heuristic}-LB^{CPLEX}}{LB^{CPLEX}}$ (%)
1	18.65
2	3.50
3	7.85
4	1.45
Overall mean	9.47

Table 6 Average deviation per product and process structure and demand profile

Product/process structure	CV	$\frac{UB^{heuristic}-LB^{CPLEX}}{LB^{CPLEX}}$ (%)
Assembly/non-cyclic	0.2	2.71
Assembly/non-cyclic	0.5	7.42
Assembly/non-cyclic	0.8	4.68
Assembly/cyclic	0.2	2.66
Assembly/cyclic	0.5	7.59
Assembly/cyclic	0.8	5.73
General/non-cyclic	0.2	10.94
General/non-cyclic	0.5	9.51
General/non-cyclic	0.8	18.74
General/cyclic	0.2	12.65
General/cyclic	0.5	9.54
General/cyclic	0.8	21.69
Overall mean		9.47

the remaining capacity profiles include utilizations up to 90%. Table 5 shows that the heuristic has a superior performance when the setup/holding cost ratio differs among the products (TBO profiles 2 and 4) compared to identical TBOs for all products (TBO profiles 1 and 3). Finally, its appears that the heuristic performs better for assembly structures than for general product structures.

Industrial data. The second set of data was taken from a company producing hair care products. There are two production stages. In a first processing step a fluid (gel, creme) is mixed in large tanks. Then on the second stage it is filled into cans or bottles. The end product is defined by the content and size of the can or bottle.

The product structure is shown in Fig. 3, where each node represents a fluid (F) or a finished product (P). All fluids are produced on resource M (mixer). The finished products are produced on resource L (filling line). The problem data are outlined in Table 7.

A planning horizon of 70 periods was considered. For the given data, we tried to solve model MLCLSP-L on the same Unix workstation as above. After 12 h of CPU time, CPLEX found a solution within a MIPGAP (maximum allowed relative difference between upper and lower bound) of 6% (the originally targeted MIPGAP of 5% was not reached within 72 h). Within 63.72 s, the proposed heuristic (run on the PC) found a feasible solution, which is 0.58% below the solution found by CPLEX. Thus, for the considered industrial case the heuristic provides a solution which is at most 5.42% above the optimum.

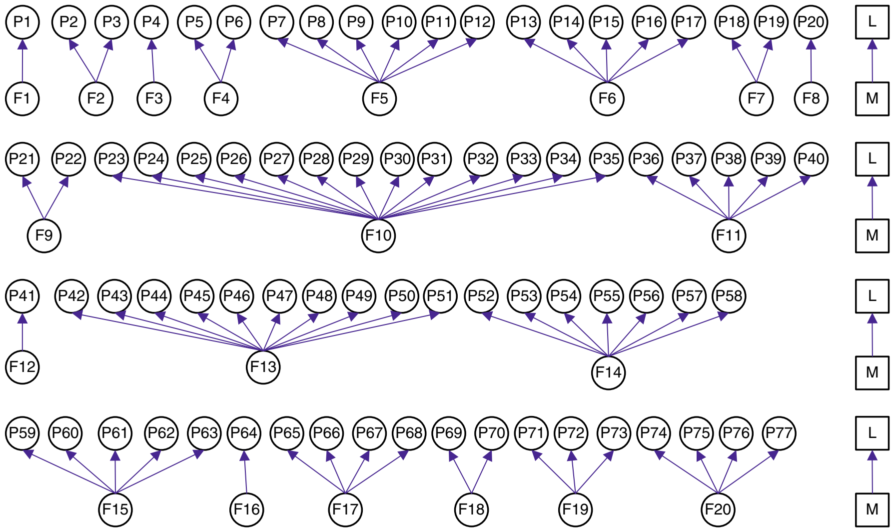

Fig. 3 Product structure for the industrial example

Table 7 Data for the industrial example

	Finished products	Fluids
Number of items	77	20
Setup costs	37.51	225.06
Holding costs (per week and unit)	0.00595 – 0.02237	0.005
Production time (minutes per unit)	0.02082 – 0.02562	0
Setup time (min)	15	90

5 Conclusions and future work

We have formulated an extension of the well-known dynamic multi-level lotsizing model (MLCLSP) that also accounts for setup carryovers. For the solution of this model we proposed a Lagrangean heuristic. The quality of the heuristic has been tested based on a data set with 600 small problem instances from the literature, a new data set containing 1,920 problem instances with increased size, and with data taken from industry. It appears that the heuristic is able to generate solutions with good quality. However, it must be noted that due to the complexity of the problem the computation of benchmark results is extremely time-consuming. Future works will be devoted to broadening the computational basis of the numerical evaluation.

Appendix

We will follow Billington et al. (1983, 1986) and Tempelmeier and Derstroff (1996) in eliminating the inventory variable y_{kt} from the model. The inventory of item k at the end of period t is equal to the initial inventory plus cumulative production less cumulative demand:

$$y_{kt} = \sum_{\tau=1}^{t} (q_{k\tau} - d_{k\tau}) - \sum_{j \in \mathcal{S}_k} \sum_{\tau=0}^{t} a_{kj} \cdot q_{j,\tau+1} + \widehat{y}_k \qquad k = 1, 2, \ldots, K; \quad t = 1, \ldots, T \tag{36}$$

The total holding costs are then

$$\sum_{k=1}^{K} \sum_{t=1}^{T} h_k \cdot \left(\sum_{\tau=1}^{t} (q_{k\tau} - d_{k\tau}) - \sum_{j \in \mathcal{S}_k} \sum_{\tau=0}^{t} a_{kj} \cdot q_{j,\tau+1} + \widehat{y}_k \right),$$

which can be rewritten as the sum of

$$\sum_{k=1}^{K} \sum_{t=1}^{T} h_k \cdot \left(\widehat{y}_k - \sum_{\tau=1}^{t} d_{k\tau} \right), \tag{37}$$

which is a constant, and

$$\sum_{k=1}^{K} \sum_{t=1}^{T} h_k \cdot \left(\sum_{\tau=1}^{t} q_{k\tau} - \sum_{j \in \mathcal{S}_k} \sum_{\tau=0}^{t} a_{kj} \cdot q_{j,\tau+1} \right) \tag{38}$$

Rearranging (38) leads to

$$\sum_{k=1}^{K} \sum_{t=1}^{T} h_k \cdot \left(\sum_{\tau=1}^{t} q_{k\tau} - \sum_{j \in \mathcal{S}_k} \sum_{\tau=1}^{t+1} a_{kj} \cdot q_{j\tau} \right)$$
$$= \sum_{k=1}^{K} \sum_{t=1}^{T} \sum_{\tau=1}^{t} \left(h_k \cdot q_{k\tau} - \sum_{j \in \mathcal{S}_k} h_k \cdot a_{kj} \cdot q_{j\tau} \right) - \sum_{k=1}^{K} \sum_{j \in \mathcal{S}_k} \sum_{t=1}^{T} \left(h_k \cdot a_{kj} \cdot q_{j,t+1} \right)$$

As $q_{T+1} = 0$, this is equal to

$$\sum_{k=1}^{K} \sum_{t=1}^{T} \sum_{\tau=1}^{t} \left(h_k \cdot q_{k\tau} - \sum_{j \in \mathcal{S}_k} h_k \cdot a_{kj} \cdot q_{j\tau} \right) - \sum_{k=1}^{K} \sum_{j \in \mathcal{S}_k} \sum_{t=2}^{T} \left(h_k \cdot a_{kj} \cdot q_{jt} \right)$$

Changing from the perspective of the predecessor to that of the successor results in

$$= \sum_{k=1}^{K} \sum_{t=1}^{T} \sum_{\tau=1}^{t} q_{k\tau} \cdot \left(h_k - \sum_{j \in \mathcal{P}_k} h_j \cdot a_{jk} \right) - \sum_{k=1}^{K} \sum_{t=2}^{T} \left(\sum_{j \in \mathcal{P}_k} h_j \cdot a_{jk} \right) \cdot q_{kt} \tag{39}$$

Let e_k be the marginal holding cost coefficient for item k defined as

$$e_k = h_k - \sum_{j \in \mathcal{P}_k} a_{jk} \cdot h_j.$$

Then Eq. (39) dissolves to

$$\sum_{k=1}^{K}\sum_{t=1}^{T}\sum_{\tau=1}^{t} e_k \cdot q_{k\tau} - \sum_{k=1}^{K}\sum_{t=2}^{T} (h_k - e_k) \cdot q_{kt} \tag{40}$$

The second term $(-\sum_{k=1}^{K}\sum_{t=2}^{T} (h_k - e_k) \cdot q_{kt})$ is subtracted, as the production of a successor item reduces the inventory of its components one period earlier than it adds to its own inventory. Throughout the lead time, which is one period, the respective quantity of the components is thus regarded as work in process inventory and does not incur holding costs.

Note that the only reason for the earlier production of predecessor items is that derived demand arises before the end of a period. This does not only effect a reduction of holding costs in Eq. (40), but also an increase in holding costs caused by initial inventory (see Eq. (37)) and earlier production of predecessors.

Throughout the planning horizon, cumulative production equals cumulative total (external and derived) demand. Thus, when there is no inventory at the end of the planning horizon, we have

$$\sum_{t=1}^{T} q_{kt} = \sum_{t=1}^{T} D_{kt}, \tag{41}$$

With positive initial inventory, cumulative total demand is computed as follows:

$$D_{kt} = \max\left\{0, d_{kt} + \sum_{j \in N_k} a_{kj} \cdot D_{j,t+1} - \max\left\{0, \widehat{y}_k - \sum_{\tau=0}^{t-1}\left(d_{k\tau} + \sum_{j \in N_k} a_{kj} \cdot D_{j,\tau+1}\right)\right\}\right\} \tag{42}$$

Using (41), the second (cost reduction) term in (40) can be rearranged to

$$\begin{aligned}\sum_{k=1}^{K}\sum_{t=2}^{T}(h_k - e_k) \cdot q_{kt} &= \sum_{k=1}^{K}(h_k - e_k) \cdot \left(\sum_{t=1}^{T} D_{kt} - q_{k1}\right)\\ &= \sum_{k=1}^{K}\sum_{t=1}^{T}(h_k - e_k) \cdot D_{kt} - \sum_{k=1}^{K}(h_k - e_k) \cdot q_{k1}.\end{aligned} \tag{43}$$

Again, the first part of Eq. (43) is constant and the second variable. Adding the former to (37) yields

$$\begin{aligned}&\sum_{k=1}^{K}\sum_{t=1}^{T} h_k \cdot \left(\widehat{y}_k - \sum_{\tau=1}^{t} d_{k\tau}\right) - \sum_{k=1}^{K}\sum_{t=1}^{T}(h_k - e_k) \cdot D_{kt}\\ &= -\sum_{k=1}^{K}\sum_{t=1}^{T} h_k \cdot ((T - t + 1) \cdot d_{kt} - \widehat{y}_k) - \sum_{k=1}^{K}\sum_{t=1}^{T}(h_k - e_k) \cdot D_{kt}.\end{aligned} \tag{44}$$

Replacing the cost reduction component in Eq. (39) by the variable part of Eq. (43) yields

$$\sum_{k=1}^{K}\left(\sum_{t=1}^{T} e_k \cdot (T-t+1) \cdot q_{kt} + (h_k - e_k) \cdot q_{k1}\right) \tag{45}$$

Using (44) and (45) to replace holding costs and Eq. (36) to replace the inventory variable, the objective function is

$$Z = \sum_{k=1}^{K}\sum_{t=1}^{T}(s_k \cdot \gamma_{kt} + e_k \cdot (T-t+1) \cdot q_{kt}) + \sum_{k=1}^{K}(h_k - e_k) \cdot q_{k1} - \mathrm{C} \tag{46}$$

with

$$\mathrm{C} = \sum_{k=1}^{K}\sum_{t=1}^{T} h_k \cdot ((T-t+1) \cdot d_{kt} - \widehat{y}_k) + \sum_{k=1}^{K}\sum_{t=1}^{T}(h_k - e_k) \cdot D_{kt}. \tag{47}$$

References

Billington PJ, McClain JO, Thomas LJ (1983) Mathematical programming approaches to capacity-constrained mrp systems: review, formulation and problem reduction. Manage Sci 29(10):1126–1141

Billington PJ, McClain JO, Thomas LJ (1986) Heuristics for multilevel lot-sizing with a bottleneck. Manage Sci 32(8):989–1006

Briskorn D (2006) A note on capacitated lot sizing with setup carry over. IIE Trans 38:1045–1047

Derstroff MC (1995) Mehrstufige Losgrößenplanung mit Kapazitätsbeschränkungen. Produktion und Logistik. Physica-Verlag, Heidelberg

Drexl A, Kimms A (1997) Lot sizing and scheduling - survey and extensions. Eur J Oper Res 99(2):221–235

Haase K (1994) Lotsizing and scheduling for production planning. Springer, Berlin

Haase K (1998) Capacitated lot-sizing with linked production quantities of adjacent periods. In: Drexl A, Kimms A (eds.) Beyond manufacturing resource planning (MRP II) – advanced models and methods for production planning. Springer, Berlin pp 127–146

Jans R, Degraeve Z (2004) Meta-heuristics for dynamic lot sizing: a review and comparison of solution approaches. ERIM Report Series Research in Management pp 1–38

Karimi B, Ghomi SMTF, Wilson JM (2003) The capacitated lot sizing problem: a review of models and algorithms. Omega 31:365–378

Meyr H (1999) Simultane Losgrößen- und Reihenfolgeplanung für kontinuierliche Produktionslnien. Gabler Edition Wissenschaft: Produktion und Logistik. Gabler, Wiesbaden

Salomon M (1991) Deterministic lotsizing for production planning, Band 355 of lecture notes in economics and mathematical systems. Springer, Berlin

Sox CR, Gao Y (1999) The capacitated lot sizing problem with setup carry-over. IIE Trans 31:173–181

Sürie C (2005) Time continuity in discrete time models. Springer, Berlin

Staggemeier AT, Clark AR (2001) A survey of lot-sizing and scheduling models. In: 23rd annual symposium of the brazilian operational research society (SOBRAPO). Campos do Jordao SP, Brazil, pp 938–947

Tempelmeier H, Derstroff M (1996) A lagrangean-based heuristic for dynamic multilevel multi-item constrained lotsizing with setup times. Manag Sci 42(5):739–757

Wagner HM, Whitin TM (1958) Dynamic version of the economic lot size model. Manag Sci 5(1):89–96