4OR (2008) 6:61–83
DOI 10.1007/s10288-007-0057-1

RESEARCH PAPER

Capacitated lot-sizing with extensions: a review

Daniel Quadt · Heinrich Kuhn

Received: 17 March 2006 / Revised: 13 September 2007 / Published online: 30 October 2007

Abstract The capacitated lot-sizing problem (CLSP) is a standard formulation for big bucket lot-sizing problems with a discrete period segmentation and deterministic demands. We present a literature review on problems that incorporate one of the following extensions in the CLSP: back-orders, setup carry-over, sequencing, and parallel machines. We illustrate model formulations for each of the extensions and also mention the inclusion of setup times, multi-level product structures and overtime in a study. For practitioners, this overview allows to check the availability of successful solution procedures for a specific problem. For scientists, it identifies areas that are open for future research.

Keywords Capacitated lot-sizing · Back-orders · Setup carry-over · Sequencing · Parallel machines

MSC classification (2000) 90B30 · 90B35 · 90C10 · 90C27 · 90C59 · 90C90

1 Introduction

We address the capacitated lot-sizing problem (CLSP) and give a state-of-the-art literature review with respect to the following extensions: parallel machines, back-orders, setup carry-over, and sequencing. The latter two include scheduling decisions into the lot-sizing problem.

D. Quadt · H. Kuhn (✉)
Department of Production, Logistics and Operations Management,
Catholic University of Eichstätt-Ingolstadt, Auf der Schanz 49, 85049 Ingolstadt, Germany
e-mail: heinrich.kuhn@ku-eichstaett.de

D. Quadt
e-mail: daniel.quadt@ku-eichstaett.de

The (standard) CLSP can be described as follows: multiple products have to be produced. A deterministic, discrete demand volume for every product is given for predefined periods. Producing a product consumes machine capacity, which is scarce. When changing from one product to another, setup costs accrue. In some studies, setup times are also incurred, reducing machine capacity. When a product unit is not produced in its demand period, product specific inventory costs are incurred. The objective is to find an optimal production plan that minimizes setup and inventory costs and delivers optimal lot-sizes and production periods for each product.

The extensions of the CLSP covered in this paper have been selected due to their importance in the industrial practice. The extension of *parallel machines* implies that there is no unambiguous product-to-machine assignment. Instead, a product may be produced on any of the parallel machines. This enlarges the planning problem, as a decision has to be made on which machine to produce a product unit and how many machines to use in parallel for each product in each period ('loading' or 'assignment' problem). Traditional approaches that use a single, aggregated machine will generally lead to inferior solutions for the parallel machine case, because only a single setup would be counted for the aggregate resource if the calculated volume has to be produced on more than one machine. In this case, in reality, several setups have to be performed in parallel with no savings occurring compared with a lot-for-lot policy. Parallel machine problems can be found in a large number of industries, such as the chemical, electronics, food and textile industries (see, e.g., Wittrock 1988; Riane 1998; Moursli and Pochet 2000).

The inclusion of *setup carry-over* means that a machine is able of carrying over its setup state between period boundaries. While the standard CLSP implies a setup for each product produced per period (and machine), a setup carry-over implies that the last product per period may be produced without an additional setup in the subsequent period. Haase (1998) points out that solutions become significantly different when setup carry-over is considered. In addition, if setup carry-over is accounted together with parallel machines, a lot-for-lot policy could substantially reduce the number of setup operations. In the extreme case, when a machine is occupied by one product during the entire planing horizon, there would not be a setup operation at all. The possibility of carrying over a setup between periods is given in many industries, for example in the semiconductor industry, where production runs 24 h a day and 7 days a week (see Quadt and Kuhn 2005). Any superimposed period boundary without the possibility of carrying over the setup state would result in lost capacity, which is a crucial factor in capital intensive industries such as the semiconductor industry.

The possibility of carrying over a setup state enlarges the problem by a scheduling decision, as, for each machine, a decision has to be made which product shall be the first and the last in a period. Another extension is to include scheduling decisions for all products, and thus to determine a total *sequence* of all products on each machine. This allows to include sequence-dependent setup costs and times. Sequence-dependent setup costs and times can be found in many industries, for example the chemical industry, where it is crucial to find the right product sequence to optimize capacity utilization (e.g., Cooke and Rohleder 2006).

Finally, *back-orders* broaden the scope of the problem by allowing a product to be produced after the given demand period. In this case, back-order costs accrue for every period and unit of the delay. In highly capacitated environments as well as in many real-life situations, the inclusion of back-orders is crucial because otherwise, no feasible plan would exist—and the respective result that no feasible solution can be found is of minor importance in practical settings. In fact, the question of interest is which products shall be back-ordered and which not.

The outline of this paper is as follows: In Sect. 2, we present a mixed integer programming (MIP) model formulation of the standard CLSP. Section 3 is devoted to the literature review, beginning with studies and procedures covering back-orders (Sect. 3.1), followed by setup carry-over (Sect. 3.2), sequencing (Sect. 3.3), and parallel machines (Sect. 3.4). We also present model formulations that cover these extensions. A summary including two tables that classify the discussed studies and procedures together with a table with model sizes finally completes the survey in Sect. 4.

2 Model formulation

Lot-sizing models can be classified as small bucket or big bucket models. Small bucket models consist of relatively short periods. They usually allow only one product or setup per period and machine. Big bucket models contain fewer but longer periods and usually have no restriction on the number of products or setups per period and machine. Standard big bucket models do not consider the product sequence within a period. Hence, a production schedule cannot immediately be read from a solution. With small bucket models, a production schedule can directly be concluded from a solution, as the sequence of products is given by the sequence of periods and the products produced in those periods.

This general advantage of small bucket models arises at the cost of higher computation times. For the same planning horizon, the number of periods must be substantially larger than in big bucket models to yield comparable solutions. We focus on the CLSP (a big bucket model) and its extensions. Formulations of the standard CLSP can for example be found in Billington et al. (1983) or Trigeiro et al. (1989). We present a model formulation that includes setup costs and setup times. We will include setup carry-over (also commonly referred to as 'linked lot-sizes'), sequencing decisions, back-orders and parallel machines in the respective sections of the literature review.

Parameters

c_p^i	Inventory holding costs of product p
c_p^s	Setup costs of product p
C_t	Capacity in period t
d_{pt}	Demand volume of product p in period t
P	Number of products, $\bar{P} = \{1 \dots P\}$
t_p^s	Setup time of product p
t_p^u	Process time per unit of product p
y_p^0	Initial inventory volume of product p at the beginning of the planning interval
T	Number of periods, $\bar{T} = \{1 \dots T\}$
z	Big number, $z \geq \sum_{p \in \bar{P},\, t \in \bar{T}} d_{pt}$

Variables

x_{pt} Production volume of product p in period t

y_{pt} Inventory volume of product p at the end of period t

γ_{pt} Binary setup variable for product p in period t ($\gamma_{pt} = 1$, if a setup is performed for product p in period t)

Model CLSP (capacitated lot-sizing problem)

$$\min \sum_{\substack{p \in \bar{P} \\ t \in \bar{T}}} \left(c^i_p y_{pt} + c^s_p \gamma_{pt} \right) \tag{2.1}$$

subject to

$$y_{p,t-1} + x_{pt} - d_{pt} = y_{pt} \quad \forall p \in \bar{P},\ t \in \bar{T} \tag{2.2}$$

$$\sum_{p \in \bar{P}} \left(t^u_p x_{pt} + t^s_p \gamma_{pt} \right) \leq C_t \quad \forall t \in \bar{T} \tag{2.3}$$

$$x_{pt} \leq z\gamma_{pt} \quad \forall p \in \bar{P},\ t \in \bar{T} \tag{2.4}$$

$$y_{p0} = y^0_p \quad \forall p \in \bar{P} \tag{2.5}$$

$$x_{pt} \geq 0 \quad \forall p \in \bar{P},\ t \in \bar{T} \tag{2.6}$$

$$y_{pt} \geq 0 \quad \forall p \in \bar{P},\ t \in \bar{T} \tag{2.7}$$

$$\gamma_{pt} \in \{0,\ 1\} \quad \forall p \in \bar{P},\ t \in \bar{T} \tag{2.8}$$

The objective function (2.1) minimizes inventory and setup costs. Equation (2.2) depicts inventory flow conditions: the demand volume of a period t must be met by inventory volume from previous periods or by a production quantity in t. Surplus volume will be stored in inventory for the next period. Condition (2.3) limits the time for processing and setups to the machine capacity. Condition (2.4) ensures that item p can only be produced in a period t if a setup for p is performed in t. Equation (2.5) sets the initial inventory volumes and conditions (2.6)–(2.8) enforce non-negative, respectively, binary variables. The CLSP contains PT binary variables, $2PT$ continuous variables and $5PT + P + T$ constraints.

Bitran and Yanasse (1982) show that the CLSP is NP-hard even without setup times (i.e., with setup costs only), meaning that one cannot expect to find an efficient algorithm generating an optimal solution. When setup times are included as in the formulation above, Maes et al. (1991) show that the feasibility problem already becomes NP-complete. This implies that one cannot efficiently say whether a feasible solution exists at all. Hence, heuristics are needed that do not guarantee an optimal solution, but find a reasonably good solution in a moderate amount of computation time. This holds true especially when real world problems are to be solved, as the computation time increases with the problem size. We will give an overview of such solution procedures.

3 Literature review

Lot-sizing problems have been addressed extensively in the literature. An overview of (mainly) small bucket models and solution procedures can be found in Drexl and Kimms (1997). Research integrating lot-sizing and scheduling decisions in small and big bucket models is reviewed by Meyr (1999).

Various formulations of the big bucket CLSP, lower bounding techniques, and solution approaches including extensions to multi-level product structures are well known (e.g., Kleindorfer and Newson 1975; Eppen and Martin 1987; Trigeiro et al. 1989; Millar and Yang 1993; Tempelmeier and Helber 1994, 1995; Tempelmeier and Derstroff 1996; Stadtler 1996; Katok et al. 1998; Miller et al. 2000; Özdamar and Bozyel 2000).

In addition, several linkages to other specialized OR problems are documented, e.g., the relation to the facility location problem (Maes et al. 1991), the network flow problem (Eppen and Martin 1987) and the Steiner tree problem with and without hop constraints (Erickson et al. 1987; Voss 1999).

Reviews and taxonomies of such models and solution procedures are given by Bahl et al. (1987), Maes and Van Wassenhove (1988), Kuik et al. (1994), Helber (1994) and Derstroff (1995). A recent in-depth review by Karimi et al. (2003) summarizes research on the general capacitated lot-sizing problem. We focus on studies considering CLSP-like models (i.e., capacitated big bucket models with a discrete period segmentation) that include one or more of the following extensions: (1) back-orders, (2) setup carry-over, (3) sequencing or (4) parallel machines. The inclusion of setup times, multi-level product structures and overtime is mentioned as well, even though the literature on these topics is not covered exhaustively. We give a short summary of the developed solution approaches and cite the sizes of solved test problems in order to give an indication of the computation times of the procedures.

3.1 Back-order literature

Back-orders can be included in the CLSP by adding the following parameters and variables:

Parameters

b_p^0 Initial back-order volume of product p at the beginning of the planning interval

c_p^b Back-order costs of product p

Variables

b_{pt} Back-order volume of product p at the end of period t

The objective function is augmented by back-order costs:

$$\min \sum_{\substack{p \in \bar{P} \\ t \in \bar{T}}} \left(c_p^i y_{pt} + c_p^b b_{pt} + c_p^s \gamma_{pt} \right) \tag{3.1}$$

The inventory flow conditions are adjusted so that the demand volume of a period t plus the back-order volume of previous periods must be met by inventory volume from previous periods or by a production quantity in t. If the demand volume cannot be met, it will be back-ordered to the next period while surplus volume will be stored in inventory:

$$y_{p,t-1} - b_{p,t-1} + x_{pt} - d_{pt} = y_{pt} - b_{pt} \quad \forall p \in \bar{P},\ t \in \bar{T} \tag{3.2}$$

In addition, the new Eq. (3.3) ensure that the whole demand volume is produced until the end of the planning horizon. Further, the back-order variables have to be initialized using condition (3.4) and restricted to non-negative values with condition (3.5):

$$b_{pT} = 0 \quad \forall p \in \bar{P} \tag{3.3}$$

$$b_{p0} = b_p^0 \quad \forall p \in \bar{P} \tag{3.4}$$

$$b_{pt} \geq 0 \quad \forall p \in \bar{P},\ t \in \bar{T} \tag{3.5}$$

We call the new model CLSP-BO (CLSP with back-orders). It includes the objective function (3.1) and conditions (2.3)–(2.8) as well as (3.2)–(3.5). The CLSP-BO comprises PT binary variables, $3PT$ continues variables and $6PT+3P+T$ constraints.

Despite its importance in practical settings, only few researchers have addressed capacitated lot-sizing problems with back-ordering. Smith-Daniels and Smith-Daniels (1986) present a single machine mixed integer programming model with back-orders that is a combination of a small bucket and a big bucket approach. In each period, only one product family can be produced. A family may consist of various items. While there are sequence-dependent setup times between items, no setup time is incurred between families as family setups are supposed to take place between periods. On the other hand, there are sequence-independent family setup costs but no item-to-item setup costs. As usual in small bucket models, a family can carry over its setup to a subsequent period. A simplified version of the model including four items and five periods is solved using standard procedures. In Smith-Daniels and Ritzman (1988), the model is extended to the multi-level case. In this version, more than one family can be produced per period, making it a big bucket model. Now, a sequence-dependent setup time is incurred for family changeovers, but item-to-item setup times do not accrue. Setup costs are not considered at all. The approach automatically sequences the families within a period and is able of carrying over a family setup from one period to another. A single instance consisting of three families with two items each, three periods and two stages is solved using standard procedures.

Pochet and Wolsey (1988) tackle the single machine CLSP with back-orders using a shortest-path and an alternative plant location formulation. The first one is solved by standard mixed integer programming procedures, the second by a cutting plane algorithm. Setup times and setup carry-over are not considered. They solve problems of up to 100 products and 8 periods.

Millar and Yang (1994) extend an approach from Millar and Yang (1993) and present two algorithms for a single machine lot-sizing problem with back-orders. One algorithm is based on Lagrangian decomposition, the other on Lagrangian relaxation, both neither covering setup times nor setup carry-over. Their test problems consist of five products and six periods.

Cheng et al. (2001) consider the analogies between the CLSP with back-orders and the traditional fixed charge transportation problem. Their solution procedure is three-phased. After heuristically determining setups for some products in certain periods, they use a procedure similar to Vogel's approximation method for the transportation problem to create an initial solution. Afterwards, they employ a variation of the standard primal transportation simplex to improve the solution. Despite covering only a single machine, their model takes different kinds of capacity sources into consideration. Only a single setup has to be performed when one or more of the capacity sources are used in a period, nevertheless, the production costs differ between capacity sources. This allows the modeling of regular and overtime. Setup times and setup carry-over are not considered. Cheng et al. (2001) solve problems with 3 products and 12 periods.

Hung and Chien (2000) argue that, since the feasibility problem for the CLSP with non-zero setup times is NP-complete (see Maes et al. 1991), a lot of effort may be used to determine the feasibility of the problem. However, knowing the feasibility status of the problem does not help a production planner at all. In most cases, it is impossible to satisfy all orders in time. In this situation a planner wants to satisfy all confirmed orders as on-time as possible by minimizing the sum of back-order, inventory, and setup costs. Hung and Chien (2000), therefore, extend the classical MLCLSP with back-orders and allow setup times as well as setup costs. Since solving the MIP formulation of realistic problems by an optimal procedure is likely to be impractical, they suggest and analyze three metaheuristics (Tabu Search, Simulated Annealing, and a Genetic Algorithm). The metaheuristics find integer solutions for the setup variables, whereas the evaluation of the solution is performed by solving an LP problem. Their test problems are randomly generated with up to 10 periods, 20 items, 5 end-products and 3 stages, each stage with a single machine. The results show that Tabu Search and Simulated Annealing perform best.

3.2 Setup carry-over literature

The following parameters, variables and conditions include setup carry-over (or 'linked lot-sizes', see Haase 1994) in the CLSP:

Parameters

ζ_p^0 Initial setup state of product p at the beginning of the planning interval ($\zeta_p^0 = 1$, if the machine is initially set up for product p)

Variables

ζ_{pt} Binary linking variable for product p in period t ($\zeta_{pt} = 1$, if the setup state for product p is carried over from period t to $t+1$)

$$x_{pt} \leq z\left(\gamma_{pt} + \zeta_{p,t-1}\right) \qquad \forall p \in \bar{P},\ t \in \bar{T} \tag{3.6}$$

$$\sum_{p \in \bar{P}} \zeta_{pt} = 1 \qquad \forall t \in \bar{T} \tag{3.7}$$

$$\zeta_{pt} - \gamma_{pt} - \zeta_{p,t-1} \leq 0 \qquad \forall p \in \bar{P},\ t \in \bar{T} \tag{3.8}$$

$$\zeta_{pt} + \zeta_{p,t-1} - \gamma_{pt} + \gamma_{qt} \leq 2 \quad \forall p, q \in \bar{P},\ q \neq p,\ t \in \bar{T} \tag{3.9}$$

$$\zeta_{p0} = \zeta_p^0 \quad \forall p \in \bar{P} \tag{3.10}$$

$$\zeta_{pt} \in \{0,\ 1\} \quad \forall p \in \bar{P},\ t \in \bar{T} \tag{3.11}$$

Condition (3.6) replaces condition (2.4). They link production to the correct setup state: item p can only be produced in a period t if either the machine carries over a setup state for p from period $t-1$ (in which case $\zeta_{p,t-1} = 1$), or a setup for p is performed in period t ($\gamma_{pt} = 1$). Conditions (3.7)–(3.9) handle the correct implementation of the setup carry-over. Condition (3.7) implies that the machine can only carry over a setup for one product. Condition (3.8) states that a setup carry-over is possible only if the machine has been set up for the product. That is, if the machine carries over a setup state for product p from period t to $t+1$ ($\zeta_{pt} = 1$), there must be a setup for p in period t ($\gamma_{pt} = 1$) or the setup state has been carried over from the previous period ($\zeta_{p,t-1} = 1$). Condition (3.9) deals with the following situation: If the machine carries over a setup state for item p from period $t-1$ to t and also from t to $t+1$ ($\zeta_{p,t-1} = \zeta_{pt} = 1$) and, in period t, a setup is performed for another product q ($\gamma_{qt} = 1$), then we have to re-set up the machine to p in period t ($\gamma_{pt} = 1$). Condition (3.10) initializes the setup carry-over variables and condition (3.11) defines them as binary variables.

The model CLSPL (CLSP with linked lot-sizes) consists of the objective function (2.1) and conditions (2.2)–(2.3), (2.5)–(2.8) and (3.6)–(3.11). It has $2PT$ binary variables, $2PT$ continues variables and $PPT + 6PT + 2P + 2T$ constraints.

Dillenberger et al. (1993) cover the CLSP with setup carry-over and without back-ordering. Their research is motivated by a practical production planning problem and is slightly different from the one usually found in the lot-sizing literature: production of a product must always take place in its demand period. If capacity does not allow the complete demand to be met, products are back-logged. The difference between back-logging and back-ordering is that back-ordered product units *are produced* in a period after their demand, while back-logged units *are not produced* at all and the demand is lost. The approach covers setup times and parallel machines. Dillenberger et al. (1994) extend the model to different kinds of resources like energy or components. Some resources are storable, meaning if their capacity is not consumed in a period, it may be used in later periods. Products are grouped into product families. Switching from family to family incurs a major setup, while changing products within a family incurs a minor setup, leading to partially sequence-dependent setups. Both major and minor setups may incur setup times and costs. The solution procedure consists of a partial Branch and Bound (B&B) scheme that iterates on a period-by-period basis. In each iteration, only setup variables relating to the actual period are fixed to binary values, while the binary restrictions for later periods are relaxed. The authors solve practical problems with up to 40 products, 6 periods and 15 machines.

Haase (1994) establishes the name CLSPL for a CLSP with setup carry-over. He solves a standard CLSPL with a single machine and without back-ordering. Setup times are not considered. His procedure moves backwards from the last to the first period and schedules setups and production volumes using a randomized regret measure. He points out that, from a conceptual point of view, a backward-oriented procedure is superior when solving a standard CLSPL: a forward-oriented procedure must decide how many units to produce for future demands, depending on complicated and time-consuming estimations of future capacity availabilities. Without back-orders, a backward-oriented procedure can always focus on the actual period as remaining unmet demands of later periods are already known. Clearly, with back-orders allowed, this observation is obsolete, as a backward-oriented procedure would have to decide how many units to back-order just as a forward-oriented procedure must decide how many units to store in inventory. Haase (1994) solves well known problems from Thizy and Van Wassenhove (1985) including eight products and eight periods. Later, Haase (1998) reduces the model formulation so that a setup state cannot be preserved over more than one period boundary. The resulting model is much easier to solve and the solution quality is similar—given his test problems covering a single machine. He solves problems with 50 products and 8 periods, 20 products and 20 periods as well as 8 products and 50 periods.

Hindi (1995b) considers a specialized single-item, dynamic lot-sizing (Wagner–Whitin) problem. Hindi assumes a capacitated production facility where a startup cost is incurred for switching the production facility on and a separate reservation cost is incurred for keeping the facility on wether it is used for production or not in the next period. The formulated problem of finding the least cost schedule for a single-item arises as a subproblem in a multi-product CLSPL, where reservation costs represent Lagrange multipliers. Thus, efficient solution procedures are necessary for the single-item case. Hindi suggests a fast Tabu Search solution procedure. Problems with up to 20 periods are tested. Even though the Tabu Search procedure cannot guarantee an optimal solution, for all instances tested, optimal solutions are found.

Gopalakrishnan et al. (1995) present a new mixed integer programming model for a variation of the CLSPL covering parallel machines but no back-ordering. Setup times and costs are included, but are product-independent—meaning they are the same for all products. The formulation makes extensive use of binary variables in comparison to other formulations. They solve a practical problem covering 12 periods, 3 machines and 2 product families using standard procedures. Gopalakrishnan (2000) considers a slightly different problem. In this study, setup times and costs are product-dependent. On the other hand, the new model formulation only includes a single machine. A single instance with two products and three periods is solved to optimality using standard procedures. Gopalakrishnan et al. (2001) develop a Tabu Search algorithm for this problem and solve problems with 30 products and 20 periods.

Sox and Gao (1999) develop a new model for the CLSPL. Their formulation covers a single machine, no back-ordering and no setup times. It differs from the approach by Dillenberger et al. (1994) and Haase (1994), as it does not contain a binary variable indicating that a product is the only product being produced in a specific period (on a certain machine). A shortest-path reformulation of the model is solved to optimality for smaller test instances covering eight items and eight periods. For larger problems, the setup carry-over is restricted to one subsequent period per product as suggested by Haase (1998). Using a Lagrangian decomposition heuristic, instances with up to 100 products and 10 periods are solved to near-optimality in less than 1 min computation time on a Sparc workstation 20. Gao (2000) proposes a simple forward period-by-period heuristic for the same restricted problem. Test problems with 50 products and 8 periods, respectively, 20 products and 20 periods, indicate that the new heuristic delivers solutions with higher costs but is much faster.

Sürie and Stadtler (2003) (see also Sürie 2005) use a simple plant location reformulation of the CLSPL as introduced by Haase (1994). Their approach covers setup times and multiple machines in a multi-level production environment, but the product/machine assignment is unique, meaning that the machines cannot be used in parallel. Back-ordering is not allowed. They employ a Branch and Cut (B&C) and a Cut and Branch (C&B) algorithm, which add valid inequalities to the model and solve the test instances described by Gopalakrishnan et al. (2001). For larger instances of the test, they suggest a time-oriented decomposition approach that moves a lot-sizing time-window through the planning periods. In each of its iterations, the problem is solved for all periods, but the complete set of constraints is only enforced for the periods of the time-window. Periods before the time-window have been planned in an earlier iteration and their binary variables are fixed. In the time interval following the lot-sizing window, capacity utilization is only estimated by neglecting setups.

Lot-sizing models with setup carry-over determine the last product in each period. Thus, they establish a partial order of the products. Some researches have developed models and procedures that construct a total sequence of all products in big bucket models. The last product of the sequence can carry over its setup to the next period. As such, these approaches also deal with setup carry-over. They will be presented in the following section.

3.3 Sequencing literature

The CLSD (capacitated lot-sizing problem with sequence-dependent setups) is a variation of the CLSPL. It establishes a complete order of all products produced in a period. As its name implies, the purpose is to include sequence-dependent setup costs and times. We use the following notation:

Parameters

c^s_{qp}	Setup costs *from* product q *to* product p
t^s_{qp}	Setup time *from* product q *to* product p

Variables

γ_{qpt}	Binary setup variable indicating a setup from product q to product p in period t ($\gamma_{qpt} = 1$, if a setup is performed from product q to product p in period t)
π_{pt}	Sequencing variable indicating the ordinal position of product p in period t. The larger π_{pt}, the later product p is scheduled in t

The objective function (2.1) as well as conditions (2.3) and (3.6) of the CLSPL are slightly modified so that setups are taken into account for all potential preceding products q (by using the sum of all associated variables $\gamma_{qpt}\ \forall q \in \bar{P}$). The adjusted objective function is shown in (3.12), the adjusted conditions in (3.13)–(3.14):

$$\min \sum_{\substack{p \in \bar{P} \\ t \in \bar{T}}} c^i_p y_{pt} + \sum_{\substack{p,\, q \in \bar{P} \\ t \in \bar{T}}} c^s_p \gamma_{qpt} \tag{3.12}$$

$$\sum_{p \in \bar{P}} \left(t_p^u x_{pt} + \sum_{q \in \bar{P}} t_p^s \gamma_{qpt} \right) \leq C_t \quad \forall t \in \bar{T} \tag{3.13}$$

$$x_{pt} \leq z \left(\zeta_{p,t-1} + \sum_{q \in \bar{P}} \gamma_{qpt} \right) \quad \forall p \in \bar{P},\ t \in \bar{T} \tag{3.14}$$

Further, the setup carry-over conditions (3.8) and (3.9) of the CLSPL are replaced by conditions (3.15) and (3.16), and conditions (3.17) are added (see Haase 1996):

$$\sum_{q \in \bar{P}} \gamma_{qpt} + \zeta_{p,t-1} = \sum_{r \in \bar{P}} \gamma_{prt} + \zeta_{pt} \quad \forall p \in \bar{P},\ t \in \bar{T} \tag{3.15}$$

$$\pi_{pt} \geq \pi_{qt} + 1 - P\left(1 - \gamma_{qpt}\right) \quad \forall p, q \in \bar{P},\ t \in \bar{T} \tag{3.16}$$

$$\pi_{pt} \geq 0 \quad \forall p \in \bar{P},\ t \in \bar{T} \tag{3.17}$$

Equation (3.15) establishes a kind of setup flow. If the machine is set up to product p in period t or a setup carry-over from the previous period is used for p, then the machine must be set up from p to (another) product r or carry over the setup state for p to the next period. This implies that every product—with the exception of the first and the last product per period—has a predecessor and a successor. Condition (3.16) generates the sequence of products per period and thus eliminate sub-tours. The larger π_{pt}, the later product p will be scheduled in period t. Whenever a setup from product q to product p is performed ($\gamma_{qpt} = 1$), the expression $P\left(1 - \gamma_{qpt}\right)$ equals zero, and thus $\pi_{pt} \geq \pi_{qt} + 1$ must hold true. This implies that p is scheduled after q. Condition (3.16) also prevents a setup to the same product ($\gamma_{ppt} \neq 1$ for all $p \in \bar{P},\ t \in \bar{T}$). Finally, the sequence variables are limited to non-negative values by (3.17). They automatically become integers through condition (3.16).

The CLSD consists of the objective function (3.12) and conditions (2.2), (2.5)–(2.8), (3.7), (3.10)–(3.11) and (3.13)–(3.17). The CLSD contains $PPT + PT$ binary variables, $3PT$ continues variables and $PPT + 8PT + 2P + 2T$ constraints.

Aras and Swanson (1982) develop a backward-oriented period-by-period loading procedure for a single machine problem with sequence-independent setup times. As setup costs are not considered, the objective is to minimize total holding costs. Holding costs are taken into account even within the period of production. Therefore, a total sequence of all products is generated and the machine can carry over its setup state to the next period. Back-orders are not allowed. They solve a practical problem covering 38 products and 26 periods.

Selen and Heuts (1990) discuss a lot-sizing and scheduling problem in a chemical manufacturing environment where production lots must constitute an integer number of pre-defined quantities (batches). The problem covers a single machine. In addition to standard problems, the inventory capacity is limited. Setup times are sequence-dependent and setup carry-overs are possible. Back-orders are not allowed. The proposed solution procedure starts with a feasible production plan based on a lot-for-lot solution. It shifts complete production lots to earlier periods, with the target period moving forward in a period-by-period manner. A sample problem with 20 products and 10 periods is solved. Heuts et al. (1992) compare two modified versions of the described heuristic in a rolling horizon setting. One modification moves the source period for a production shift instead of the target period. In the other modification, all periods are eligible as source and target period at once. Both procedures lead to similar results in different computational experiments with 15 products and 34 periods (out of which 10 are considered in each run due to the rolling horizon).

Haase (1996) formulates the CLSD as presented above. It contains a single machine and setup times while back-orders are not considered. He uses a variation of the backward-oriented, randomized regret based heuristic described by Haase (1994) to solve the problem. The test instances cover up to 9 products and 8 periods or 4 products and 26 periods. Haase and Kimms (2000) present a new model that incorporates sequence-dependent setup times. For each period, the model has to select a sequence of products from an a priori given set of such sequences and to determine the production volumes. In its essence, the solution procedure is a B&B method. It moves backwards from the last period to the first one, selecting one of the given sequences in each iteration. The size of the instances solved ranges from 3 products and 15 periods to 10 products and 3 periods.

Grünert (1998) presents a multi-machine model for the multi-level case. However, a unique assignment of products to machines must be given and hence the machines cannot be used in parallel. The model contains sequence-dependent setup times and an option to use overtime, but back-orders are not allowed. Tabu Search is employed in conjunction with Lagrangian decomposition. The generation of neighboring solutions during the Tabu Search procedure is guided by the solution of the Lagrangian problem. When the latter suggests a setup for a product in a certain period and the setup is not tabu, it will be included in the new candidate solution. Problems with up to 300 products and 10 periods or 10 products and 30 periods are tested, all with one machine per production level. The number of production levels in the test problems is a result of the stochastic instance generator and can therefore not be ascertained. Nevertheless, Grünert (1998, p. 259) mentions that computation times become prohibitive when more than ten products have to be planned on a single machine.

Fleischmann and Meyr (1997) develop a model that can be viewed as both a small bucket and a big bucket model: Each ‘macro-period’ consists of (possibly empty) ‘micro-periods’ of variable length. While the number of products per micro-period is restricted to one, the number of products per macro-period is potentially unbounded. The model contains a single machine and establishes a total order of all products. Setup times and back-orders are not considered. They suggest a solution method based on Threshold Accepting. Starting from an infeasible solution, it generates new setup sequences by performing neighborhood operations such as inserting a setup for a certain product, exchanging two setups or deleting one. Each setup sequence is evaluated using a heuristic procedure that determines the production and inventory volumes. Computational tests are performed using the instances from Haase (1996). Meyr (2000) states that once the setup sequences are fixed, the remaining problem of determining production and inventory volumes becomes a linear programming (LP) problem. He drops the heuristic sub-procedure and replaces it with an efficient network flow algorithm to evaluate the setup sequences optimally. The algorithm exploits the fact that the setup sequences differ only marginally and, thus, a solution can be found using re-optimization. Besides Threshold Accepting, he also uses Simulated

Annealing as a coordinating procedure and extends the model to sequence-dependent setup times. In addition to a comparison with the results of Haase (1996), he solves problems with setup times covering up to 18 products and 8 periods.

Laguna (1999) introduces another model. It also contains a single machine and sequence-dependent setup times, and extends the problem by an overtime option. Again, back-orders are not allowed. A solution approach similar to the one by Meyr (2000) is suggested. The method starts with an LP-relaxation of the model and a following Traveling Salesman algorithm to order the products of each period. Production and inventory volumes of an initial solution are determined by solving another mixed integer program. Afterwards, a Tabu Search procedure is invoked to improve the solution by local search. Each solution is evaluated by using the Traveling Salesman procedure to re-sequence the products and the mixed integer program to find production and inventory volumes. Both these sub-problems are solved to optimality for each candidate by using standard procedures. Test instances cover 3 products and 12 periods or 5 products and 3 periods.

3.4 Parallel machines literature

Parallel machines can be added to the CLSP by augmenting the production variables x_{pt} and the capacity parameter C_t by an additional index m indicating the individual machines. In the following, we present a formulation including back-orders and setup carry-over, thus comprising some of the previous extensions in one formulation. This implies that also the variables γ_{pt} and ζ_{pt} and parameters ζ_p^0 have to be augmented by a machine index m. The complete notation is given as follows:

Parameters

b_p^0	Initial back-order volume of product p at the beginning of the planning interval
c_p^b	Back-order costs of product p
c_p^i	Inventory holding costs of product p
c_p^s	Setup costs of product p
C_{tm}	Capacity of machine m in period t
d_{pt}	Demand volume of product p in period t
M	Number of parallel machines, $\bar{M} = \{1 \dots M\}$
P	Number of products, $\bar{P} = \{1 \dots P\}$
t_p^s	Setup time of product p
t_p^u	Per unit process time of product p
y_p^0	Initial inventory volume of product p at the beginning of the planning interval
T	Number of periods, $\bar{T} = \{1 \dots T\}$
z	Big number, $z \geq \sum_{p \in \bar{P},\, t \in \bar{T}} d_{pt}$
ζ_{pm}^0	Initial setup state of product p on machine m at the beginning of the planning interval ($\zeta_{pm}^0 = 1$, if machine m is initially set up for product p)

Variables

b_{pt}	Back-order volume of product p at the end of period t
x_{ptm}	Production volume of product p in period t on machine m
y_{pt}	Inventory volume of product p at the end of period t

γ_{ptm} Binary setup variable for product p in period t on machine m ($\gamma_{ptm} = 1$, if a setup is performed for product p in period t on machine m)

ζ_{ptm} Binary linking variable for product p in period t on machine m ($\zeta_{ptm} = 1$, if the setup state for product p on machine m is carried over from period t to $t + 1$)

Model CLSPL-BOPM (CLSP with linked lot-sizes, back-orders and parallel machines)

$$\min \sum_{\substack{p \in \bar{P} \\ t \in \bar{T}}} c_p^i y_{pt} + \sum_{\substack{p \in \bar{P} \\ t \in \bar{T}}} c_p^b b_{pt} + \sum_{\substack{p \in \bar{P} \\ t \in \bar{T} \\ m \in \bar{M}}} c_p^s \gamma_{ptm} \tag{3.18}$$

subject to

$$y_{p,t-1} - b_{p,t-1} + \sum_{m \in \bar{M}} x_{ptm} - d_{pt} = y_{pt} - b_{pt} \qquad \forall p \in \bar{P},\ t \in \bar{T} \tag{3.19}$$

$$\sum_{p \in \bar{P}} \left(t_p^u x_{ptm} + t_p^s \gamma_{ptm} \right) \leq C_{tm} \qquad \forall t \in \bar{T},\ m \in \bar{M} \tag{3.20}$$

$$x_{ptm} \leq z \left(\gamma_{ptm} + \zeta_{p,t-1,m} \right) \qquad \forall p \in \bar{P},\ t \in \bar{T},\ m \in \bar{M} \tag{3.21}$$

$$\sum_{p \in \bar{P}} \zeta_{ptm} = 1 \qquad \forall t \in \bar{T},\ m \in \bar{M} \tag{3.22}$$

$$\zeta_{ptm} - \gamma_{ptm} - \zeta_{p,t-1,m} \leq 0 \qquad \forall p \in \bar{P},\ t \in \bar{T},\ m \in \bar{M} \tag{3.23}$$

$$\zeta_{ptm} + \zeta_{p,t-1,m} - \gamma_{ptm} + \gamma_{qtm} \leq 2 \qquad \forall p, q \in \bar{P},\ q \neq p,\ t \in \bar{T},\ m \in \bar{M} \tag{3.24}$$

$$b_{pT} = 0 \qquad \forall p \in \bar{P} \tag{3.25}$$

$$b_{p0} = b_p^0,\ y_{p0} = y_p^0,\ \zeta_{p0m} = \zeta_{pm}^0 \qquad \forall p \in \bar{P},\ m \in \bar{M} \tag{3.26}$$

$$b_{pt} \geq 0,\ x_{ptm} \geq 0,\ y_{pt} \geq 0,\ \gamma_{ptm} \in \{0, 1\},\ \zeta_{ptm} \in \{0, 1\} \qquad \forall p \in \bar{P},\ t \in \bar{T},\ m \in \bar{M} \tag{3.27}$$

The CLSPL-BOPM consists of $2PTM$ binary variables, $PTM + 2PT$ continues variables and $PPTM + 4PTM + 3PT + PM + TM + 3P + T$ constraints. Without the extensions of setup carry-over and back-orders, the CLSP with parallel machines (CLSP-PM) contains PTM binary variables, $PTM + PT$ continuous variables and $3PTM + 2PT + TM + P$ constraints. The explosion of the number of binary variables makes the extension with parallel machines a very difficult problem when the number of parallel machines is large. However, many potential solutions are symmetrical to each other (i.e., they can be transformed into each other by swapping machines). Hence, it is crucial for a solution procedure to efficiently break the symmetry of the parallel machine problem.

Besides the articles by Dillenberger et al. (1993, 1994) and Gopalakrishnan et al. (1995), which have already been mentioned above, little research is available that integrates parallel machines into big bucket lot-sizing models.

Diaby et al. (1992) include parallel resources in the CLSP. Their formulation contains setup times, but no setup carry-over and back-orders. Only a single setup has to be performed whenever a product is produced in a period, no matter how many resources are being used. While this assumption is valid when parallel resources are employed to represent regular and overtime or similar concepts, it is not suited for the parallel machines concept used here. Using a Lagrangian relaxation scheme with subgradient optimization, they solve instances with up to 5,000 products and 30 periods.

Derstroff (1995) solves a multi-level parallel machine CLSP, also using a Lagrangian relaxation procedure. In a first step, the model is solved with relaxed capacity and inventory flow constraints. In a second step, the Lagrangian multipliers are updated, and, in a third step, a feasible solution is created by shifting production volumes to different periods and machines. While setup times are covered, setup carry-over and back-ordering are not considered. The test instances include 20 products on 5 production levels, 16 periods and 6 machines, of which up to two can be used in parallel.

Hindi (1995a) considers a parallel machine lot-sizing problem *without* setups. Back-orders are not allowed. The problem is reformulated as a capacitated transshipment model and is solved by using a primal and a dual network simplex algorithm. Instances with up to 200 items, 24 periods and 2 machines are solved on a Sun ELC workstation.

Özdamar and Birbil (1998) tackle a parallel machine lot-sizing problem with setup times but without setup costs. In addition to regular capacity, each machine is allowed to use a certain amount of overtime capacity at respective costs. Back-ordering and setup carry-over are not allowed. The parallel machines concept differs from the one described here as lot-splitting is not allowed, i.e., the complete per-period production volume of a product must be produced on a single machine. In other words, only one of the parallel machines can be used to produce a specific product per period. Three similar hybrid heuristics are developed to solve the problem. In each of them, a Genetic Algorithm operates mainly in the infeasible region of the solution space. An interweaved Tabu Search and Simulated Annealing algorithm is used to improve single solutions of a population and make them feasible. Sample problems with up to 20 products, 6 periods and 5 machines are solved. Özdamar and Barbarosoglu (1999) extend the problem to multiple production stages. They also include setup times and the possibility to back-order end-products. Lot splitting remains prohibited. A stand-alone Simulated Annealing procedure is compared with two hybrid heuristics combining Simulated Annealing with a Genetic Algorithm (in a manner similar to the above hybrid heuristics) and a Lagrangian relaxation approach, respectively. The latter provides better results with respect to solution quality and computation time for test problems with up to 20 products, 6 periods and 5 machines on each of the 4 production stages.

Kang et al. (1999) present a model and solution procedure tailor-made for the so-called 'CHES problems' described by Baker and Muckstadt (1989). The model establishes a total order of all products on each machine, allowing the setup state to

Table 1 Model sizes of the CLSP and extensions

Model	Binary variables	Continuous variables	Constraints
CLSP	PT	$2PT$	$5PT + P + T$
CLSP-BO	PT	$3PT$	$6PT + 3P + T$
CLSPL	$2PT$	$2PT$	$PPT + 6PT + 2P + 2T$
CLSD	$PPT + PT$	$3PT$	$PPT + 8PT + 2P + 2T$
CLSP-PM	PTM	$PTM + PT$	$3PTM + 2PT + TM + P$
CLSP-BOPM	$2PTM$	$PTM + 2PT$	$PPTM + 4PTM + 3PT + PM + TM + 3P + T$

M number of parallel machines, P number of products, T number of periods

be carried over to a subsequent period. Setup times and back-orders are not included. The problems are based on real-life cases and cover up to 21 products, 3 periods and 10 machines. However, only two of the five problems cover more than one period, and those include only one or two machines. Unlike usual lot-sizing problems, the objective is to maximize the profit contribution: products are sold at market prices and the given demand may be exceeded. Kang et al. (1999) extend the test-bed by instances covering six products, nine periods and one or two machines. The problems are solved by a truncated B&B procedure that incorporates a Column Generation approach for the LP-relaxations at each node. Afterwards, the solution is improved using a hill climbing method.

Belvaux and Wolsey (2000) describe a generic modelling and optimization system which is able to solve a wide class of lot-sizing problems, including special cases of multi-item, multi-machine, multi-level, multi-period capacitated lot-sizing problems with both big and small bucket model formulations. With respect to the lot-sizing extensions surveyed in this review, back-orders and parallel machines could be considered, whereas the setup carry-over cannot be handled. The formulated lot-sizing problems are solved by XPRESS-MP B&B routines, a lot-sizing specific preprocessing, and the generation of problem oriented cutting planes. In addition, a primal heuristic may be integrated into the solution procedure. The authors solve a large number of test instances all taken from the literature, including the above mentioned 'CHES problems' described by Baker and Muckstadt (1989). Four out of these five test instances are solved to optimality, whereas the procedure of Kang et al. (1999) solves two instances to optimality only.

Karimi et al. (2006) extend the classical CLSPL with back-orders. They develop a heuristic procedure which starts with an initial feasible solution. This solution is improved by adopting the corresponding setup and setup carry-over schedule and re-optimizing it by solving a minimum-cost network flow problem. Then the improved solution is used as a starting solution for a Tabu Search procedure. Their test problems are randomly generated with 4–12 products and 4–8 periods. Only small test instances (4 products, 4 or 6 periods) are solved to optimality or near-optimality, respectively.

Meyr (2002) extends the algorithm described by Meyr (2000) to work with parallel machines. Again, he uses a Threshold Accepting and a Simulated Annealing

Table 2 Literature review of capacitated big bucket lot-sizing models with respect to back-ordering, setup times, setup carry-over and parallel machines. x: Covered in the reference; (x) and ((x)): partly covered in the reference, see text

References	Back-orders	Setup times	Setup carry-over	Sequencing	Parallel machines	Over-time	Multi-level
Part 1							
Smith-Daniels and Smith-Daniels (1986)	x	(x)	x	(x)			
Smith-Daniels and Ritzman (1988)	x	x	x	x			x
Pochet and Wolsey (1988)	x						
Millar and Yang (1994)	x						
Cheng et al. (2001)	x					x	
Hung and Chien (2000)	x	x					x
Dillenberger et al. (1993)		x	x		x		
Dillenberger et al. (1994)		x	x		x		
Haase (1994)			x				
Haase (1998)			x				
Hindi (1995b)			x				
Gopalakrishnan et al. (1995)		(x)	x		x		
Gopalakrishnan (2000)		x	x				
Gopalakrishnan et al. (2001)		x	x				
Sox and Gao (1999)			x				
Gao (2000)			x				
Sürie and Stadtler (2003)		x	x				x
Aras and Swanson (1982)		x	x	x			
Selen and Heuts (1990)		x	x	x			
Heuts et al. (1992)		x	x	x			
Haase (1996)			x	x			
Haase and Kimms (2000)		x	x	x			
Grünert (1998)		x	x	x		x	x
Fleischmann and Meyr (1997)			x	x			
Meyr (2000)		x	x	x			
Laguna (1999)		x	x	x		x	
Part 2							
Diaby et al. (1992)		x			((x))	x	
Derstroff (1995)		x			x		x
Hindi (1995a)					x		
Özdamar and Birbil (1998)		x			(x)	x	
Özdamar and Barbarosoglu (1999)	x	x			(x)	x	x
Kang et al. (1999)			x	x	x		
Belvaux and Wolsey (2000)	x	x			x		x
Karimi et al. (2006)	x		x				
Meyr (1999)		x	x	x	x		
Meyr (2002)		x	x	x	x		
Quadt and Kuhn (2004)	x	x	x		x		
Quadt (2004)	x	x	x		x		

Table 3 Abbreviations

Model	
CLSP	Capacitated lot-sizing problem
CLSPL	Capacitated lot-sizing problem with linked lot-sizes (setup carry-over)
CLSD	Capacitated lot-sizing problem with sequence-dependent setups
-BO	With back-orders
-PM	With parallel machines
Solution approaches	
B&B	Branch and Bound
CP	Cutting plane
Decom	Decomposition
GA	Genetic algorithm
Heu	Heuristic
HC	Hill climbing
LO	Lagrangian optimization
LP	Linear programming
MIP	Mixed integer programming
SA	Simulated Annealing
TA	Threshold Accepting
TS	Tabu Search

approach to fix the setup sequence and a network flow algorithm to determine the production and inventory volumes. However, compared with the single machine case, the network flow algorithm becomes more complicated and time-consuming when non-identical machines are considered. Meyr (2002) evaluates the approach with test instances covering 15–19 products, 8 periods and 2 machines. A modified version of the heuristic is also compared with the one of Kang et al. (1999). Meyr (1999) suggests a decomposition approach for larger problems, based on the above algorithms. The procedure is explained using a real-world case, but no numerical tests are given.

Quadt and Kuhn (2004) and (2005) formulate the CLSPL-BOPM as presented in this paper. It covers back-orders, setup times, setup carry-over and identical parallel machines. They solve the problem using a novel model re-formulation that uses integer instead of binary variables to break the symmetry of the parallel machines. A period-by-period heuristic iteratively solves instances of the new model and generates a schedule. The authors solve problems with up to 20 products, 6 periods and 10 machines. Quadt (2004) shows that the procedure is able to handle a large number of parallel machines efficiently. He solves problems with up to 100 machines (with 6 products and 6 periods) as well as problems with up to 6 products, 15 periods and 6 machines.

Table 4 CLSP with extensions: applied solution approaches and largest problem size of test instances. A blank entry in the respective column indicates that multi-level product structures or parallel machines are not considered

References	Problem	Algorithm	Products	Periods	Stages	Machines per stage
Part 1						
Smith-Daniels and Smith-Daniels (1986)	CLSP-BO	MIP	4	5		
Smith-Daniels and Ritzman (1988)	CLSP-BO	MIP	6	3	2	
Pochet and Wolsey (1988)	CLSP-BO	MIP-CP	100	8		
Millar and Yang (1994)	CLSP-BO	LO	5	6		
Cheng et al. (2001)	CLSP-BO	Heu-LP	3	12		
Hung and Chien (2000)	CLSP-BO	Heu-TA, -SA, -GA	3–5	10	3	
Dillenberger et al. (1993)	CLSPL-PM	MIP-Decom		No tests		
Dillenberger et al. (1994)	CLSPL-PM	MIP-Decom		No tests		
Haase (1994)	CLSPL	Heu	8	8		
Haase (1998)	CLSPL	Heu	50	8		
			20	20		
			8	50		
Hindi (1995b)	CLSPL	Heu-TS	1	20		
Gopalakrishnan et al. (1995)	CLSPL-PM	MIP	2	12		3
Gopalakrishnan (2000)	CLSPL	MIP	2	3		
Gopalakrishnan et al. (2001)	CLSPL	Heu-TS	30	20		
Sox and Gao (1999)	CLSPL	MIP	8	8		
		Heu-LO	100	10		
Gao (2000)	CLSPL	Heu	50	8		
		Heu-LO	20	20		
Sürie and Stadtler (2003)	CLSPL	MIP-CP, Heu	30	20		
Karimi et al. (2006)	CLSPL-BO	Heu-TS	10	8		
			12	6		
Aras and Swanson (1982)	CLSD	Heu	38	26		
Selen and Heuts (1990)	CLSD	Heu	20	10		
Heuts et al. (1992)	CLSD	Heu	15	34		
Haase (1996)	CLSD	Heu	9	8		
			4	26		
Haase and Kimms (2000)	CLSD	Heu-B&B	3	15		
			10	3		
Grünert (1998)	CLSD	Heu-TS	300	10		
			10	30		
			6	15		6
Part 2						
Fleischmann and Meyr (1997)	CLSD	Heu-TA	9	8		
			4	26		
Meyr (2000)	CLSD	Heu-TA, -SA	18	8		

Table 4 continued

References	Problem	Algorithm	Products	Periods	Stages	Machines per stage
Laguna (1999)	CLSD	Heu-TS	3	12		
			5	3		
Diaby et al. (1992)	CLSP	Heu-LO	5,000	30		
Derstroff (1995)	CLSP-PM	Heu-LO	20	16	5	2 of 6
Hindi (1995a)	CLSP-PM	Heu-LP	200	24		2
Özdamar and Birbil (1998)	CLSP-PM	Heu-GA, -TS, -SA	20	6		1 of 5
Özdamar and Barbarosoglu (1999)	CLSP-PM	Heu-GA, -SA	20	6	4	1 of 5
Kang et al. (1999)	CLSP-PM	Heu-B&B-HC	21	3		10
			6	9		2
Belvaux and Wolsey (2000)	CLSP-BOPM	MIP-CP	21	3		10
			6	9		2
Meyr (1999)	CLSD-PM	Heu-Decom		No tests		
Meyr (2002)	CLSD-PM	Heu-TA, -SA	15–19	8		2
Quadt and Kuhn (2004)	CLSPL-BOPM	Heu-MIP	20	6		10
Quadt (2004)	CLSPL-BOPM	Heu-MIP	100	6		6
			6	15		6

4 Summary

We have given a literature review on capacitated big bucket lot-sizing problems and solution procedures that extend the standard CLSP formulation by (1) back-orders, (2) setup carry-over, (3) sequencing or (4) parallel machines. Back-orders imply that a product volume may be produced after its demand period. Setup carry-over means that a machine does not need to be re-set up if a product is produced over period boundaries. Sequencing implies the generation of a total order of all products per machine. Parallel machines extend the problem by a loading decision as the production volumes have to be assigned to one or more of the parallel machines. Table 1 summarizes the model sizes of the CLSP and the extensions.

We have further mentioned the inclusion of setup times, overtime and multi-level product structures in a study. The findings are summarized in Table 2. Table 4 give an overview of the applied solution approaches, with abbreviations listed in Table 3.

The overview allows practitioners to check the availability of successful solution procedures for a specific problem. At the same time, it identifies areas that are open for further research.

Acknowledgments This research was supported by grant 03KUM1EI of the Bundesministerium für Bildung und Forschung (BMBF), Germany. In addition, the authors would like to thank two anonymous referees and the editor, Professor Dr. Denis Bouyssou, for their helpful comments which significantly improved the paper.

References

Aras OA, Swanson LA (1982) A lot sizing and sequencing algorithm for dynamic demands upon a single facility. J Oper Manage 2(3):177–185

Bahl HC, Ritzman LP, Gupta JND (1987) Determining lot sizes and resource requirements: a review. Oper Res 35(3):329–345

Baker T, Muckstadt J Jr (1989) The CHES problems. Working paper, Chesapeake Decision Sciences, Inc., NJ

Belvaux G, Wolsey LA (2000) bc – prod: a specialized branch-and-cut system for lot-sizing problems. Manage Sci 46(5):724–738

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

Bitran GR, Yanasse HH (1982) Computational complexity of the capacitated lot size problem. Manage Sci 28(10):1174–1186

Cheng CH, Madan MS, Gupta Y, So S (2001) Solving the capacitated lot-sizing problem with backorder consideration. J Oper Res Soc 52:952–959

Cooke DL, Rohleder TR (2006) Inventory evaluation and product slate management in large-scale continuous process industries. J Oper Manage 24(3):235–249

Derstroff MC (1995) Mehrstufige Losgrößenplanung mit Kapazitätsbeschränkungen. Physica, Heidelberg

Diaby M, Bahl HC, Karwan MH, Zionts S (1992) A lagrangean relaxation approach for very-large-scale capacitated lot-sizing. Manage Sci 38(9):1329–1340

Dillenberger C, Escudero LF, Wollensak A, Zhang W (1993) On solving a large-scale resource allocation problem in production planning. In: Fandel G, Gulledge T, Jones A (eds) Operations research in production planning and control. Springer, Berlin pp 105–119

Dillenberger C, Escudero LF, Wollensak A, Zhang W (1994) On practical resource allocation for production planning and scheduling with period overlapping setups. Eur J Oper Res 75(2):275–286

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

Eppen GD, Martin RK (1987) Solving multi-item capacitated lot-sizing problems using variable redefinition. Oper Res 35(6):832–848

Erickson R, Monma C, Veinott A (1987) Send-and-split methods for minimum-concave-cost network flows. Math Oper Res 12:634–664

Fleischmann B, Meyr H (1997) The general lotsizing and scheduling problem. OR Spektrum 19:11–21

Gao Y (2000) A heuristic procedure for the capacitated lot sizing problem with set-up carry-over. Control Theory Appl 17(6):937–940

Gopalakrishnan M (2000) A modified framework for modelling set-up carryover in the capacitated lotsizing problem. Int J Prod Res 38(14):3421–3424

Gopalakrishnan M, Miller D, Schmidt C (1995) A framework for modelling setup carryover in the capacitated lot sizing problem. Int J Prod Res 33:1973–1988

Gopalakrishnan M, Ding K, Bourjolly JM, Mohan S (2001) A tabu-search heuristic for the capacitated lot-sizing problem with set-up carryover. Manage Sci 47(6):851–863

Grünert T (1998) Multi-level sequence-dependent dynamic lotsizing and scheduling. Shaker, Aachen

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

Haase K (1996) Capacitated lot-sizing with sequence dependent setup costs. OR Spektrum 18:51–59

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

Haase K, Kimms A (2000) Lot sizing and scheduling with sequence dependent setup costs and times and efficient rescheduling opportunities. Int J Prod Econ 66:159–169

Helber S (1994) Kapazitätsorientierte Losgrößenplanung in PPS-Systemen. M&P Verlag für Wissenschaft und Forschung, Stuttgart

Heuts RMJ, Seidel HP, Selen WJ (1992) A comparison of two lot sizing-sequencing heuristics for the process industry. Eur J Oper Res 59:413–424

Hindi KS (1995a) Algorithms for capacitated, multi-item lot-sizing without set-ups. J Oper Res Soc 46:465–472

Hindi KS (1995b) Solving the single-item, capacitated dynamic lot-sizing problem with startup and reservation costs by tabu search. Comput Ind Eng 28(4):701–707

Hung YF, Chien KL (2000) A multi-class multi-level capacitated lot sizing model. J Oper Res Soc 51(11):1309–1318

Kang S, Malik K, Thomas LJ (1999) Lotsizing and scheduling on parallel machines with sequence-dependent setup costs. Manage Sci 45(2):273–289

Karimi B, Fathemi Ghomi SMT, Wilson JM (2003) The capacitated lot sizing problem: a review of models and algorithms. Omega Int J Manage Sci 31(5):365–378

Karimi B, Fathemi Ghomi SMT, Wilson JM (2006) A tabu search heuristic for solving the CLSP with backlogging and set-up carry-over. J Oper Res Soc 57(2):140–147

Katok E, Lewis HS, Harrison TP (1998) Lot sizing in general assembly systems with setup costs, setup times, and multiple constrained resources. Manage Sci 44(6):859–877

Kleindorfer P, Newson E (1975) A lower bounding structure for lot-size scheduling problems. Oper Res 23(2):299–311

Kuik R, Salomon M, Van Wassenhove LN (1994) Batching decisions: structure and models. Eur J Oper Res 75(2):243–263

Laguna M (1999) A heuristic for production scheduling and inventory control in the presence of sequence-dependent setup-times. IIE Trans 31:125–134

Maes J, Van Wassenhove LN (1988) Multi-item single-level capacitated dynamic lot-sizing heuristics: a general review. J Oper Res Soc 39(11):991–1004

Maes J, McClain JO, Van Wassenhove LN (1991) Multilevel capacitated lotsizing complexity and LP-based heuristics. Eur J Oper Res 53:131–148

Meyr H (1999) Simultane Losgrößen- und Reihenfolgeplanung für kontinuierliche Produktionslinien – Modelle und Methoden im Rahmen des Supply Chain Management. Gabler, Wiesbaden

Meyr H (2000) Simultaneous lotsizing and scheduling by combining local search with dual reoptimization. Eur J Oper Res 120(2):311–326

Meyr H (2002) Simultaneous lotsizing and scheduling on parallel machines. Eur J Oper Res 139(2):277–292

Millar HH, Yang M (1993) An application of lagrangean decomposition to the capacitated multi-item lot sizing problem. Comput Oper Res 20(4):409–420

Millar HH, Yang M (1994) Lagrangian heuristics for the capacitated multi-item lot-sizing problem with backordering. Int J Prod Econ 34(1):1–15

Miller AJ, Nemhauser GL, Savelsbergh MWP (2000) On the capacitated lot-sizing and continuous 0–1 knapsack polyhedra. Eur J Oper Res 125:298–315

Moursli O, Pochet Y (2000) A branch-and-bound algorithm for the hybrid flowshop. Int J Prod Econ 64:113–125

Özdamar L, Barbarosoglu G (1999) Hybrid heuristics for the multi-stage capacitated lot sizing and loading problem. J Oper Res Soc 50:810–825

Özdamar L, Birbil SI (1998) Hybrid heuristics for the capacitated lot sizing and loading problem with setup times and overtime decisions. Eur J Oper Res 110(3):525–547

Özdamar L, Bozyel MA (2000) The capacitated lot sizing problem with overtime decisions and setup times. IIE Trans 32(11):1043–1057

Pochet Y, Wolsey L (1988) Lot size models with back-logging: strong reformulations and cutting planes. Math Program 40:317–335

Quadt D (2004) Lot-sizing and scheduling for flexible flow lines. Lecture Notes in Economics and Mathematical Systems, Springer, Berlin

Quadt D, Kuhn H (2004) Capacitated lot-sizing and scheduling with parallel machines, back-orders and setup carry-over. Working paper, Ingolstadt School of Management, Catholic University of Eichstätt-Ingolstadt, Ingolstadt, Germany

Quadt D, Kuhn H (2005) A conceptual framework for lot-sizing and scheduling of flexible flow lines. Int J Prod Res 43(11):2291–2308

Riane F (1998) Scheduling hybrid flowshops: algorithms and applications. Ph.D. thesis, Facultés Universitaires Catholiques de Mons

Selen WJ, Heuts RMJ (1990) Operational production planning in a chemical manufacturing environment. Eur J Oper Res 45:38–46

Smith-Daniels VL, Ritzman LP (1988) A model for lot-sizing and sequencing in process industries. Int J Prod Res 26:647–674

Smith-Daniels VL, Smith-Daniels DE (1986) A mixed integer programming model for lot sizing and sequencing packaging lines in the process industries. IIE Trans 18:278–285

Sox CR, Gao Y (1999) The capacitated lot sizing problem with setup carry-over. IIE Trans 31:173–181
Stadtler H (1996) Mixed integer programming model formulations for dynamic multi-item multi-level capacitated lotsizing. Eur J Oper Res 94(3):561–581
Sürie C (2005) Time continuity in discrete time models. Lecture Notes in Economics and Mathematical Systems, Springer, Berlin
Sürie C, Stadtler H (2003) The capacitated lot-sizing problem with linked lot-sizes. Manage Sci 49(8):1039–1054
Tempelmeier H, Derstroff M (1996) A lagrangean-based heuristic for dynamic multilevel multiitem constrained lotsizing with setup times. Manage Sci 42(5):738–757
Tempelmeier H, Helber S (1994) A heuristic for the dynamic multi-item multi-level capacitated lotsizing for general product structures. Eur J Oper Res 75:296–311
Tempelmeier H, Helber S (1995) Lot sizing in capacitated production planning and control systems. OR Spectrum 17(1):5–18
Thizy JM, Van Wassenhove LN (1985) Lagrangean relaxation for the multi-item capacitated lot-sizing problem: a heuristic implementation. IIE Trans 17(4):308–313
Trigeiro WW, Thomas LJ, McClain JO (1989) Capacitated lot sizing with setup times. Manage Sci 35(3):353–366
Voss S (1999) The steiner tree problem with hop constraints. Ann Oper Res 86:321–345
Wittrock RJ (1988) An adaptable scheduling algorithm for flexible flow lines. Oper Res 36(4):445–453