OR Spectrum (2013) 35:33–73
DOI 10.1007/s00291-012-0296-1

REGULAR ARTICLE

Multi-stage simultaneous lot-sizing and scheduling for flow line production

Florian Seeanner · Herbert Meyr

Published online: 27 May 2012

Abstract This paper addresses the simultaneous lot-sizing and scheduling of several products in multi-stage flow line production systems consisting of heterogeneous parallel production lines per stage. The limited capacity of the production lines may be further reduced by sequence dependent setup times. Deterministic, dynamic demand of standard products has to be met without backlogging with the objective of minimizing sequence dependent setup, holding and production costs as well as costs for external purchase, overtime, and standby. Different mixed-integer programming (MIP) model formulations are proposed and tested using a standard MIP-solver. Furthermore, construction heuristics like LP-and-Fix and Relax-and-Fix are designed and applied. The solution quality and computational performance of these approaches are examined in several test scenarios.

Keywords Simultaneous lot-sizing and scheduling · Multi-level, multi-item flow production

1 Introduction

In the last decade more and more companies have introduced advanced planning systems (APS) to extend their planning capabilities. But still many planners, especially from the consumer goods industry, complain about insufficient support by those sys-

F. Seeanner (✉)
Darmstadt University of Technology, Darmstadt, Germany
e-mail: seeanner@pscm.tu-darmstadt.de

H. Meyr
University of Hohenheim, Stuttgart, Germany
e-mail: h.meyr@uni-hohenheim.de

tems. Missing scalability of solution procedures as well as unrealistic models are supposed to be the reason for this.

In the consumer goods industry, there is typically a large number of final items to be produced. This is usually done in a two- or three-stage flow line production system like make&pack (sometimes, e.g., with an additional refinement process in between). On each stage of the system, several production lines offer—at least partially—the same services and thus can be used alternatively. These lines can be considered as single planning units that are highly utilized, in general. For this reason they represent potential bottlenecks.

Since production orders are not available on time, production is made-to-stock. And because sales are usually lost, if consumer goods are not directly available to the customer, deterministic dynamic demand forecasts are to be met without backlogging. The final items can be assigned to a few setup families. Changeovers between items of the same family incur low setup costs as well as low setup times and thus can be disregarded. By contrast, high setup costs and setup times may result from changeovers between two items of different setup families. Hence, decisions have to be made not only about the sizes of the production lots and the assignment of these lots to production lines, but also about the line-specific sequences of the lots. Since in many cases only one of these production stages represents a stationary bottleneck, the problem is usually decomposed by stages and solved consecutively. But this approach fails if—because of line- and product-specific production speeds and time-varying demand—the bottleneck shifts dynamically between the stages. In that case a simultaneous consideration of multiple production stages is needed.

This paper addresses mixed-integer-programming (MIP) models for the simultaneous lot-sizing and scheduling of multi-stage production systems.[1] According to the used time structure, those models can be roughly grouped into small time bucket (STB) models with microperiods and big time bucket (BTB) models with macroperiods. STB-models use a priori defined microperiods of short length and allow at most one setup in each period. Therefore, two important advantages arise. First, STB-models automatically determine the sequence of the lots due to the natural order of the periods and the assumption of at most one setup. Second, lead times of at least a single period, which have to be postulated in MIP-models to ensure that enough pre-products are available for the next stage, are consequently short. But there are also two main drawbacks. For sake of simplicity, the setup time is usually limited to the length of a microperiod. Thus the length of a microperiod has to be chosen carefully. If it is too short, large setup times cannot be modeled and a high number of microperiods results, which increases the complexity of and the redundancy within the model. But a longer length represents the real world problem only roughly and causes a loss of modeling detail.

These shortcomings are avoided by BTB-models which only use a small number of quite long macroperiods, thus usually not limiting the length of a setup. In general, several production lots are allowed in a single macroperiod. Typically, there are two

[1] In contrast, and mainly designed for job shops, there are also hierarchical solution procedures based on an integrated lot-sizing and scheduling model. These approaches, as proposed e.g., by Dauzère-Péres and Lasserre (1994), are not in the focus of this article.

ways to determine the sequence of the production lots in BTB-models. Either those macroperiods consist again of microperiods where the length of a microperiod is not a priori fixed, but represents a decision variable (thus leading to a higher degree of freedom). Or additional "Traveling Salesman Problem (TSP)"-constraints are needed where the TSP cuts eliminate invalid subtours from the solution space. Unfortunately however, as a consequence of the long macroperiods, in both cases unrealistically high lead times result.

Even though a couple of STB- and BTB-models already exist in the literature, they all suffer from the lacks outlined above. However, Meyr (2004) tried to combine the advantages of both model types by using a common time structure for all production lines and stages. This time structure allows to benefit from short lead times as pre-items can be available for their successors in the same (micro)period they are produced. Meyr (2004) presented the General Lot-sizing and Scheduling Problem for Multiple production Stages (GLSPMS). The author tested different model variants for their suitability in terms of modeling detail and practical relevance. But further modeling aspects like runtime performance were not studied. The "best" variant found is characterized as follows: it considers multiple products with given deterministic dynamic demand, several production stages with parallel (heterogeneous) production lines per stage, limited capacity and sequence dependent setup times. Backlogging and lost sales are not allowed. However, in order to ensure feasible solutions for every demand constellation, external purchase (instead of own production) and overtime are possible. During production breaks, the setup state can be conserved—possibly incurring standby costs, which depend on the length of the break. Thus, the overall objective is to minimize inventory holding costs, sequence dependent setup costs, production costs, as well as costs for external purchasing, overtime, and standby.

In this paper we present some improvements of this "best" variant and also examine the runtime performance of this improved variant. Furthermore, it will be tested whether the runtime can be reduced by adapting extended reformulations and introducing valid inequalities, that have proven successful for other types of lot-sizing and scheduling problems. Additionally, construction heuristics based on these different formulations are compared.

Unfortunately, the underlying paper (Meyr 2004) has only been published in German language and thus is quite unknown to the international Operations Research community. Therefore, after a brief overview of existing literature on multi-stage simultaneous lot-sizing and scheduling in Sect. 2, the above mentioned "best" variant of the GLSPMS with the new improvements and its basic idea will be described comprehensively in Sect. 3. The different reformulations and the heuristics are presented in Sect. 4. The corresponding computational results can be found in Sect. 5.

2 Literature review

As already mentioned the focus of this section is on multi-stage lot-sizing and scheduling models. More comprehensive literature reviews are given by Buschkühl et al. (2010), Drexl and Kimms (1997), Karimi et al. (2003), Quadt and Kuhn (2008) as well as Zhu and Wilhelm (2005).

In the following only a few selected single-level models are presented in order to demonstrate the development in the past. Besides, these single-level models will also help to explain the differences between microperiod- and macroperiod-based time structures, as the multi-stage models can be distinguished by these structures as well. Accordingly, we differentiate between STB- and BTB-models in the following.

The Discrete Lot-sizing and Scheduling Problem (DLSP) presented by Fleischmann (1990) is deemed to be one of the first single level STB-models. It actually allows the production of only a single item per microperiod. Furthermore, this item is either produced over the whole period or not at all (all-or-nothing assumption). Since this assumption is quite restrictive it is relaxed in the Capacitated Setup Lot-sizing Problem (CSLP) by Karmarkar and Schrage (1985) and Salomon et al. (1991). The modeling detail is further improved by the Proportional Lot-sizing and Scheduling Problem (PLSP) proposed by Drexl and Haase (1995). This model now admits two different lots per microperiod separated by a single setup.

Based on the PLSP, Haase (1994) and Kimms (1996) present multi-stage STB-models. The different production stages are synchronized via the inventory balancing constraints and lead times of at least one microperiod are necessary. Furthermore, setup times are disregarded. Stadtler (2011) proposes a variant of the PLSP, which allows "zero lead times" (i.e., pre-items are also available in the same microperiod they are produced) and sequence independent setups. He solves a real-world problem of the pharmaceutical industry with a multi-level bill of materials, but only one production stage/line. Stadtler and Sahling (2011) introduce a further PLSP based formulation with zero lead times—now for multiple stages and lines, but also with sequence independent setups. Moreover, they present a solution procedure based on Relax&Fix and Fix&Optimize for solving the new model. Another multi-level STB-model is suggested by Persson et al. (2004), who solve a real-world problem with two production stages occurring in oil industry. They also disregard sequence dependent setup times and define small time buckets, which only allow a single mode of operation per each processing unit. They propose a tabu search heuristic to solve this problem.

In general, setup times present a critical issue for the STB-models with respect to the length of a microperiod. If setup times (or minimal lot-sizes) exceed the a priori defined length, quite complex formulations are necessary as proposed by Kallrath (1999) and Suerie (2005a,b, 2006) using additional time-indexed variables. On the contrary, if the length of a microperiod is chosen too long a loss of modeling detail and thus inefficient capacity utilization might result.

In BTB-models this setup time problem does not arise because macroperiods are of sufficient size. The Capacitated Lot-sizing Problem with Sequence Dependent Setup Costs (CLSD) by Haase (1996) is a single-level BTB-model that determines the setup sequence with the help of TSP-constraints. Other examples for single-stage BTB-models are the General Lot-sizing and Scheduling Problem with sequence dependent Setup Times (GLSPST) and its extension for parallel lines (GLSPPL) introduced by Meyr (2000, 2002). In both models macroperiods consist of an a priori defined number and sequence of microperiods whose length is a decision variable. In contrast to the CLSD, which only regards sequence dependent setup costs, GLSPST and GLSPPL allow for sequence dependent setup times as well.

Grünert (1998) presents a multi-level BTB-model with single production lines on each stage. The model respects setup times as well as setup costs. It also applies the inventory balancing constraints to synchronize the production stages. The setup sequence is determined with the help of TSP-constraints. However, in order to guarantee feasible plans significant lead times of one macroperiod are needed. A similar approach is proposed by Sahling (2010), who extends the CLSD for multiple stages and parallel production lines. This problem is solved with the Fix-and-Optimize heuristic—also known as "Exchange heuristic". Sikora et al. (1996) solve a lot-sizing and scheduling problem with even five production stages and a serial product structure. Unfortunately, they do not present a mathematical formulation.

Amongst other models, a multi-level formulation with macro- and microperiods is presented by Fandel and Stammen-Hegener (2006) who try to calculate inventory holding costs more accurately. This attempt leads to a non-linear model formulation. Ferreira et al. (2009) consider a production planning problem with two stages that is motivated by the soft drinks industry. They present a relaxation approach and several heuristic strategies of the Relax-and-Fix type. Toledo et al. (2009) solve the same planning problem, however, with the help of a genetic algorithm. Araujo et al. (2007) extend the GLSPST for a second production stage and backlogging in order to solve a two-stage planning problem of a foundry. They develop a Relax-and-Fix heuristic, which uses standard branch-and-cut and different local search methods to tackle the resulting sub-problems. Transchel et al. (2011) also introduce a model based on the GLSP which is fitted to the two-stage production structure of a company in the process industry. This model and two different reformulations are analyzed in terms of computational runtime and integrality gap. A further approach is presented by Mohammadi et al. (2009). They consider a multi-level flowshop production where all machines are arranged serially. Mohammadi et al. propose synchronization constraints that prevent the production of an item on a certain machine unless it has been finished on the predecessor machine. For this purpose "shadow products" (idle time) are introduced.

Meyr (2004) introduces a GLSPPL-based model with a common time structure for all production lines on all production stages that allows for short lead times despite the realistically long setup times. Furthermore, in this paper different extensions of the base model are discussed and illustrated using two (very small) examples. He makes the point that "setup splitting" and "quantity splitting" are essential to get reasonable solutions with this model. Lang (2009, Chap. 7.2) presents an extension of Meyr (2004) (by a "state-task-network"), which also allows product substitution. In the next section, the basic ideas of Meyr (2004) and the new improvements are explained in detail to give the reader a better understanding.

3 The General Lot-sizing and Scheduling Problem for Multiple production Stages

The General Lot-sizing and Scheduling Problem for Multiple production Stages (GLSPMS) is based on microperiods allowing the production of just one single item in each microperiod. Furthermore, it uses a common time structure that will be presented next.

Fig. 1 Possible structure of a microperiod

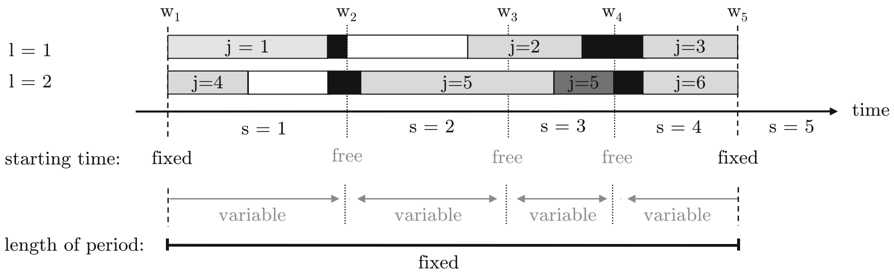

Fig. 2 Example for the common time structure

3.1 Basic idea

The idea of the common time structure is that each period s starts at the same point in time w_s for all production lines and production stages. Only the starting times of a few a priori selected periods are fixed in advance. The time span between two consecutive fixed periods can now be interpreted as (the length of) a macroperiod. That fixing allows us to model capacity,[2] demand, and holding costs based on macroperiods. The remaining non-fixed periods are called "free" periods because their starting times can be determined as decision variables of the model. As a consequence, the time span between two consecutive periods, at least one of them being free, is also a decision variable. It constitutes a microperiod of variable length.

Each microperiod can include setup, production, and standby activities, i.e., the length of a microperiod can contain shares of setup time, production time, and idle time. In order to achieve a higher capacity utilization with a smaller number of microperiods, Meyr uses "quantity splitting" and "setup splitting". Setup splitting is quite common in the literature (Drexl and Haase 1995; Haase 1994). It means that a certain part of a predefined setup time is executed at the beginning of the current period and that the remaining part is executed at the end of the directly preceding period. Analogously, quantity splitting means that the output of a part of the production time can serve as input on another production line in the same period (lead time =0), whereas the output of the remaining part can at the earliest be used by a successor line in the following period (lead time ≥1). Accordingly, as shown in Fig. 1, each microperiod may consist of six different time spans: a setup time fraction at the beginning of the microperiod, idle time before production, production being available in the same period, production being available in the subsequent period, idle time after production, and a setup time fraction at the end of the period.

Figure 2 shows a possible solution of this approach where the entire time structure can be illustrated. For instance, in Fig. 2, the starting times w_1 and w_5 of periods $s = 1$ and $s = 5$ are fixed. Accordingly, the length and thus capacity K_1 of the first macrope-

[2] Here capacity means available working time.

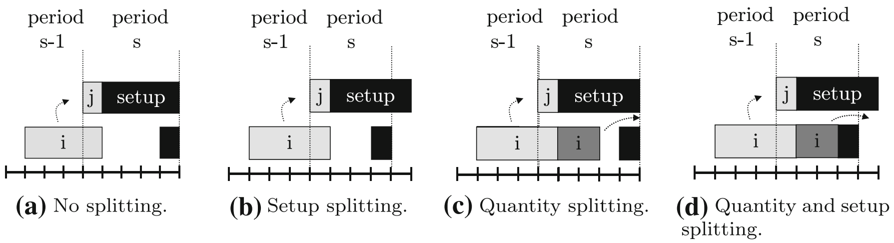

Fig. 3 Tighter plans with setup and quantity splitting

riod equals the difference of w_5 and w_1. The starting times w_2, w_3, and w_4 are a result of the optimization process. Thus, the lengths of all four microperiods belonging to the macroperiod can (almost) freely be chosen. However, their sum has to equal the length of the macroperiod. On line $l = 2$, the setup time of the first changeover is split so that a fraction belongs to microperiod $s = 1$ and the rest to $s = 2$. Moreover, on the same line in microperiod $s = 3$, the production quantity of item 5 is split. The first part is directly moved to the next stage, whereas the second part is available in the subsequent microperiod at the earliest (causing work-in-process (WIP) stock).

In order to further clarify the impact of setup and quantity splitting, Fig. 3 shows once more the mentioned enhancements. If splitting were not allowed, the result might look like Fig. 3a, where the complete changeover has to be executed within microperiod s and the production of the required pre-product i must finish on time. However, Fig. 3b–d demonstrate the improvements. In Fig. 3b, setup splitting is allowed; thus, part of the changeover can be postponed to microperiod $s + 1$ and idle time can be reduced. Quantity splitting makes it possible to keep on producing item i even if it is not needed in the same microperiod on the next stage (3c). Simultaneous setup and quantity splitting further condenses the schedule, thus leading to an even higher utilization and a more realistic model of the real world problem (3d).

Note that this common time structure allows that line synchronization can be achieved by the inventory balances per microperiod (see Eq. (4) in Sect. 3.2). Hence the significant "macroperiod lead times" of multi-stage BTB models can be eliminated. On the contrary, the time structure and the setup time splitting also allow for setup times, which can reach the length of two—already quite long—macroperiods (if the ending and starting time fractions of two subsequent microperiods consume both macroperiods' overall capacity). In addition, this time structure makes it possible to model working time calendars or non-operation periods for maintenance purposes as well.

3.2 Model formulation

In the following, we consider (physical) products $i, j = 1, \dots, J$ where $j = 1, \dots, E$ represent final items and $j = E + 1, \dots, J$ the corresponding pre-products.[3] Each product can be clearly assigned to a certain stage of the bill of materials (BOM) by

[3] Thus, the items are sorted according to their low level code in the bill of materials.

the BOM coefficients p_{ji}, denoting the quantity of predecessor item j necessary to produce one unit of successor item i. Furthermore, these items have to be scheduled on production lines $l = 1, \dots, L$ over a finite planning horizon. Note that the assignment of production lines to production stages will only indirectly be captured by the production coefficients a_{lj}, denoting the production time needed to produce one unit of item j on line l ($a_{lj} = 0$ if j cannot be produced on l).

Setup costs s_{lij} and times st_{lij} express the costs and time for changing from a product i to another product j on line l. By introducing a fictitious product 0, we are able to model a sort of "neutral" setup state, i.e., shutting down a line for costs s_{li0}, remaining idle, and starting up a line for cost s_{l0j} again. In this case, the setup state gets lost during the idle periods (without production). Note that a conservation of the last setup state during idle periods, as it was assumed in Figs. 1 and 2, can also be represented for physical products $j > 0$. This can be interpreted as holding line l in a standby mode for producing the same product again—thus incurring no additional setup costs, but possibly time-dependent standby costs b_l.

Common BTB-models work with a planning horizon T consisting of non-overlapping macroperiods $t = 1, \dots, T$. Capacity K_t, demand d_{jt}^{macro} and inventory holding costs h_{jt}^{macro} are defined per macroperiod t. According to Sect. 3.1, the same planning horizon is now additionally subdivided into S microperiods $s = 1, \dots, S$ with variable length, whose starting times are expressed by the decision variables w_s. The starting times of some of these periods, described by the set Φ, are fixed in advance. This is, for example, the case for microperiods, which mark the beginning (or end) of a macroperiod, but can also be used to set fixed downtimes of machines, e.g., during holidays or for maintenance. We introduce an artificial macroperiod $t = T + 1$, an artificial microperiod $s = S + 1$, and let f_t be the first microperiod of macroperiod t. Then the capacity K_t of a "real" macroperiod $t = 1, \dots, T$ can be expressed by the difference $w_{s''} - w_{s'}$ of the fixed starting times of two subsequent "first" microperiods $s' = f_t$ and $s'' = f_{t+1}$. Analogously, demand d_{jt}^{macro} has to be fulfilled until the end of the last microperiod of macroperiod t, which corresponds to $f_{t+1} - 1$. For ease of notation, in the following Λ denotes the set of all of these "last" microperiods. Accordingly, demand d_{js} is set to zero for all $s \notin \Lambda$ and to $d_{j,f_{t+1}-1} = d_{jt}^{\text{macro}}$, otherwise. In the same way the holding costs h_{js} can be defined on basis of macroperiods.

Furthermore, for each time interval shown in Fig. 1, decision variables have to be introduced. Accordingly, x_{ls}^{b} denotes the setup time fraction at the beginning of microperiod s on line l. Analogously, x_{ls}^{e} is defined as the setup time fraction at the end of microperiod s. The production time is computed by $a_{lj} \cdot x_{ljs}$ where x_{ljs} describes the quantity of item j produced on line l in microperiod s. Following the quantity splitting idea, x_{ljs} will further be subdivided into $\widehat{x}_{ljs}$ and $\vec{x}_{ljs}$, whose production times are the third and fourth element of Fig. 1. Finally, the idle times $\overline{x}_{ls}^{\text{b}}$ and $\overline{x}_{ls}^{\text{e}}$ represent the remaining usage (or better non-usage) of line l in microperiod s before and after the production, respectively (shutdown time for $j = 0$ and standby time for $j > 0$).

For instance, in Fig. 2, the microperiod $s = 2$ starts with idle time $\overline{x}_{12}^{\text{b}} > 0$ on line $l = 1$ and in microperiod $s = 1$ there is idle time $\overline{x}_{21}^{\text{e}} > 0$ after production on line $l = 2$. On the same line $l = 2$ item $j = 5$ is produced over the entire microperiod $s = 3$. However, this production quantity x_{253} is split into $\widehat{x}_{253}$ and $\vec{x}_{253}$ since the quantity $\vec{x}_{253}$ is not needed before microperiod $s = 4$. Besides, a splitting of the setup time occurs on line $l = 2$ in microperiods $s = 1$ and $s = 2$. Therefore, $x^{e}_{21} > 0$ and $x^{b}_{22} > 0$.

According to the introductory remarks above, we use the following notation for the model:

Indices	
i, j, v	$= 1, \ldots, J$ products, whereas =0 means neutral
k, l	$= 1, \ldots, L$ production lines (multi-stage and/or parallel)
t	$= 1, \ldots, T$ macroperiods (e.g., weeks, months)
	$= T + 1$ artificial macroperiod
s	$= 1, \ldots, S$ microperiods
	$= S + 1$ artificial microperiod
Index sets	
$\mathcal{N}_j$	Set of all direct and indirect successors of product j
$\mathcal{N}^I_j \subseteq \mathcal{N}_j$	Set of all immediate successors of product j
$\mathcal{I}_l$	Set of products that can be produced on production line l
$\mathcal{D}$	Set of all (k, i, l, j)-tuples consisting of line-product combinations (k, i) and (l, j) where product j is a direct successor of i ($j \in \mathcal{N}^I_i$) and j is producible on line l ($a_{lj} > 0$) and i on k ($a_{ki} > 0$)
Φ	Set of all microperiods with fixed starting times
Λ	Set of all last microperiods of macroperiods
Π_l	Set of all microperiods in which production on line l is not allowed
Data	
a_{lj}	Capacity consumption (time) needed to produce one unit of product j on line l
m_{lj}	Minimum lot-size of product j (units) if produced on line l
m_{l0}	Minimum time line l has to remain shut down
h_{js}	Holding costs of product j (per unit and per macroperiod t with $s = f_{t+1} - 1$)
c_{lj}	Production costs of product j (per unit) on line l
b_l	Standby costs on line l
y_{lj0}	Equals 1, if line l is set up for product j at the beginning of planning (0 otherwise)
s_{lij}	Setup costs of a changeover from product i to product j on line l
st_{lij}	Setup time of a changeover from product i to product j on production line l
d_{js}	Demand for product j in microperiod s (units)
I_{j0}	Initial inventory of product j at the beginning of planning (units)
$\overline{w}_s$	Starting time of fixed period $s \in \Phi$
p_{ji}	Number of units of product j required to produce one unit of the direct or indirect successor i
$I^{\max}$	Maximum stock level (units)
$W^{\max}_{lj}$	Maximum WIP-stock level after production on line l (units)

e_j	Purchasing costs (per unit) of product j
$e_j^{\max}$	Maximum number of units of product j that can be externally purchased
g	Overtime costs
$g^{\max}$	Maximum overtime
Variables	
$w_s \geq 0$	Starting time of microperiod s
$I_{js} \geq 0$	Inventory of product j at the end of microperiod s (units)
$x_{ls}^{\mathrm{b}} \geq 0$	Fractional setup time for changeover at the beginning of period s on line l
$x_{ls}^{\mathrm{e}} \geq 0$	Fractional setup time for changeover at the end of period s on line l
$x_{ljs} \geq 0$	Total quantity of product j produced in microperiod s on line l (units)
$\widehat{x}_{ljs} \geq 0$	Share of x_{ljs} that can be used by successors in the same microperiod s (units)
$\vec{x}_{ljs} \geq 0$	Share of x_{ljs} that can as WIP-stock first be used by successors in the following microperiod $s+1$ (units)
$\overline{x}_{ls}^{\mathrm{b}} \geq 0$	Standby time on line l in microperiod s before production
$\overline{x}_{ls}^{\mathrm{e}} \geq 0$	Standby time on line l in microperiod s after production
$o_{js} \geq 0$	Externally purchased quantity of product j in microperiod s (units)
$r_s \geq 0$	Overtime used in microperiod s
$y_{ljs} \in \{0, 1\}$	Setup state: $y_{ljs} = 1$, if line l is set up for product j in microperiod s (0 otherwise)
$z_{lijs} \geq 0$	Takes on 1, if a changeover from product i to product j takes place on line l during microperiod s (0 otherwise)

Objective function

$$\min \sum_{s\in\Lambda, j\neq 0} h_{js} I_{js} + \sum_{l,i,j,s} s_{lij} z_{lijs} + \sum_{l,j,s} c_{lj} x_{ljs} + \sum_{l,s} b_l (\overline{x}_{ls}^{\mathrm{b}} + \overline{x}_{ls}^{\mathrm{e}}) + \sum_{j\neq 0,s} e_j o_{js} + \sum_s g \cdot r_s + \sum_{l,s\in\Lambda, j\neq 0} h_{js} \vec{x}_{ljs} \tag{1}$$

subject to

$$w_s = \overline{w}_s \quad \forall s \in \Phi \tag{2}$$

$$\widehat{x}_{ljs} + \vec{x}_{ljs} = x_{ljs} \quad \forall l, j \neq 0, s \tag{3}$$

$$I_{js} = I_{j,s-1} + \sum_l \widehat{x}_{ljs} + \sum_l \vec{x}_{lj,s-1} + o_{js} - d_{js} - \sum_l \sum_{i\in N_j^I} p_{ji} x_{lis} \forall j \neq 0, s \tag{4}$$

$$I_{js} \leq I^{\max} \quad \forall j \neq 0, s \tag{5}$$

$$\vec{x}_{ljs} \leq \mathrm{W}_{lj}^{\max} \quad \forall l, j, s \tag{6}$$

$$I_{jS} = I_{j0} \quad \forall j \neq 0 \tag{7}$$

$$x_{l1}^{\mathrm{b}} = \sum_{i,j} st_{lij} z_{lij1} \quad \forall l \tag{8}$$

$$x_{l,s-1}^{\mathrm{e}} + x_{ls}^{\mathrm{b}} = \sum_{i,j} st_{lij} z_{lijs} \quad \forall l, s \geq 2 \tag{9}$$

$$x_{ls}^{\mathrm{b}} + \overline{x}_{ls}^{\mathrm{b}} + \sum_{j} a_{lj} x_{ljs} + \overline{x}_{ls}^{\mathrm{e}} + x_{ls}^{\mathrm{e}} = (w_{s+1} - w_s) + r_s \quad \forall l, s \tag{10}$$

$$a_{lj} x_{ljs} \leq \overline{w}_{S+1} y_{ljs} \quad \forall l, j, s \tag{11}$$

$$x_{ljs} \geq m_{lj}(y_{ljs} - y_{lj,s-1}) \quad \forall l, j, s \tag{12}$$

$$\sum_{j} y_{ljs} = 1 \quad \forall l, s \tag{13}$$

$$y_{ljs} = 0 \quad \forall l, j \notin \mathcal{I}_l, s \tag{14}$$

$$y_{li,s-1} + y_{ljs} - 1 \leq z_{lijs} \quad \forall l, i, j, s \tag{15}$$

$$\sum_{i,j} z_{lijs} = 1 \quad \forall l, s \tag{16}$$

$$\sum_{j \neq 0} x_{ljs} = 0 \quad \forall l, s \in \Pi_l \tag{17}$$

$$o_{js} \leq e_j^{\max} \quad \forall j \neq 0, s \tag{18}$$

$$r_s \leq g^{\max} \quad \forall s \in \Lambda \tag{19}$$

$$r_s = 0 \quad \forall s \notin \Lambda \tag{20}$$

$$x_{ls}^{\mathrm{b}} + \overline{x}_{ls}^{\mathrm{b}} \geq x_{ks}^{\mathrm{b}} + \overline{x}_{ks}^{\mathrm{b}} - \overline{w}_{S+1}(2 - y_{ljs} - y_{kis}) \quad \forall s, (k, i, l, j) \in \mathcal{D} \tag{21}$$

$$a_{ki} \widehat{x}_{kis} + \overline{x}_{ks}^{\mathrm{e}} + x_{ks}^{\mathrm{e}} \geq \overline{x}_{ls}^{\mathrm{e}} + x_{ls}^{\mathrm{e}} - \overline{w}_{S+1}(2 - y_{kis} - y_{ljs}) \forall s, (k, i, l, j) \in \mathcal{D} \tag{22}$$

The objective is to minimize the sum of holding costs of the lot-sizing stock, sequence-dependent setup costs, and production costs, as well as costs for standby, external purchase, overtime, and holding of WIP-stock (1). With the help of (2), the starting times of all microperiods in Φ (including the macroperiods) are fixed. The quantity split is allowed by (3), which divides the production quantity x_{ljs} into a part $\widehat{x}_{ljs}$ that is directly available in the same period s and into $\vec{x}_{ljs}$ that is first available in the next microperiod $s + 1$.

The inventory balancing constraints (4) ensure that primary as well as secondary demand is met without backlogging. More precisely, the inventory of a certain product j at the end of microperiod s equals the inventory of the same product at the end of the preceding microperiod plus the total inflow on stock minus the total outflow from stock during period s. The inflow on stock is composed by the total production of j during s, the WIP-stock of j that has been built up in $s - 1$, and the externally purchased quantities. Note that this WIP-stock has a lead time of a single, usually quite short microperiod and thus allows a more realistic modeling than commonly used BTB-models, as already mentioned in Sects. 1 and 2. The outflow is the primary demand of j in s and the secondary demand generated by direct successors i that are also produced in s. Constraints (5) restrict the overall inventory of each product if necessary, whereas the line-specific WIP-stock can be limited by (6). Furthermore, constraints (7) prevent zero "end-of-horizon" inventories, which may have a negative impact on later periods beyond the planning horizon (Stadtler 2000). Note that a maximum stock level for each product is necessary, if for instance the goods are perishable. A further realistic assumption might be to limit the cumulative inventory for all products. This can be easily achieved by constraints built analogously. The same is true for the WIP stock.

Constraints (8) and (9) ensure that the required setup time st_{lij} of a changeover from i to j is actually used and thus make the setup split possible. These two groups of constraints are improved as compared to Meyr (2004). First, a changeover at the beginning of microperiod $s = 1$ is now allowed. Second, the setup time fractions $x^{\mathrm{b}}_{ls}, x^{\mathrm{e}}_{ls}$ are only considered in an aggregate manner (sum over i, j) compared to the original formulation, thus reducing the number of variables. This is possible without a loss of details, since the synchronization constraints have also changed as described later on. According to the time structure outlined above, constraints (10) build up a microperiod s, which consists of setup time fractions, idle and production time. The length of this microperiod equals the time interval $w_{s+1} - w_s$ between the two subsequent microperiods s and $s + 1$, which can be prolonged if overtime r_s is used. Since the starting times of some microperiods are fixed by (2), all-in-all limited production capacity can be respected.

Because of (11) production can only take place if the line is set up accordingly. For this "setup forcing constraint", the right-hand side needs a coefficient of sufficient size. This coefficient, sometimes also denoted as "big M", has to be chosen carefully as it should not limit the production unnecessarily, but also should not be unnecessarily big. Thus the fixed end of the planning horizon $\overline{w}_{S+1}$ has been proposed.

Constraints (12) enforce a minimum lot-size and are needed because setup costs (or times) do not always satisfy the triangular inequality $s_{liv} + s_{lvj} \geq s_{lij}$.

Equation (13) enforce that a line can only be set up for exactly one product per microperiod. Since not all products can be produced on every line, (14) forbid irrelevant combinations. (15) link the setup state indicators y with the changeover indicators z. Together with the objective function (1) they ensure that z_{lijs} is only set to 1 if line l was set up for i in $s - 1$ and for j in s. Correspondingly, z_{lijs} can be defined as continuous variables. Note that since a setup for the same product j in two consecutive microperiods $s - 1$ and s is possible and since $s_{ljj} = st_{ljj} = 0$ holds for all l and j, the corresponding variables z_{ljjs} directly allow a setup carryover. That means, in contrast to other formulations like the Capacitated Lot-Sizing Problem with Linked lot sizes (CLSPL) (Haase 1994; Suerie and Stadtler 2003), no additional variables for a setup carryover are necessary. Moreover, when an idle period s without any production activities occurs on a line l ($\sum_{j>0} x_{ljs} = 0$), it is up to the model to decide whether

- the setup state should be conserved for the last product j produced on this line ($y_{lj,s-1} = 1$) without incurring any setup costs or times ($z_{ljjs} = z_{ljj,s+1} = 1$),

- a changeover to another product $i > 0, i \neq j$ should be executed ($z_{ljjs} = z_{lji,s+1} = 1$), or
- the setup state should get lost (by changing to the fictitious product $j = 0$) because the production line should be shut down after period $s - 1$ and started up again at the beginning of period $s + 1$ ($z_{lj0s} = z_{l0i,s+1} = 1$).

Note that most models for simultaneous lot-sizing and scheduling only allow modeling either a conservation or a loss of the setup state after idle periods, but not both. To get a tighter formulation, (16) are added (but not necessarily needed).

With respect to e.g., working calendars or pre-determined maintenance activities (17) allow production to be prohibited on certain lines in certain microperiods. Constraints (18), (19), and (20) impose limits on external purchasing and overtime, respectively. Note that only the capacity of fixed microperiods can be extended by overtime.

Finally, the multi-stage production enforces the additional synchronization constraints (21) and (22) to guarantee feasible plans. These constraints are new and make the major difference to Meyr (2004). When considering a predecessor line k producing a predecessor product i and a successor line l producing a (direct) successor product j, both equations ensure that the production of j must neither start before production of product i starts nor end before the (relevant) production of product i (which is needed in the same period) ends. Of course, a necessary prerequisite is that both products can be produced on the respective lines at all, i.e., that $a_{ki} > 0$ and $a_{lj} > 0$. This is expressed by the index set $\mathcal{D}$, which contains all tuples (k, i, l, j) that fulfill these conditions.

As illustrated in Fig. 4 constraints (21) force setup and idle time before production on successor line l to be at least as long as setup and idle time before production on line k. Constraints (22) assure that setup and idle time after production on line l are at most as long as the time needed on line k for producing the parts which are first available in the next microperiod, standby and setup at the end of the current microperiod. However, these conditions only make sense if pre-product i is actually produced on k and successor j on l. Accordingly, both types of constraints are only "active", if products i and j are set up on lines k and l, respectively. Since (21) and (22) prevent an incorrect timing *for each* predecessor-successor relation and since the inventory balancing constraints (4) ensure *aggregate* material availability, the resulting production schedules appear fairly realistic even for parallel (predecessor and/or successor) lines.

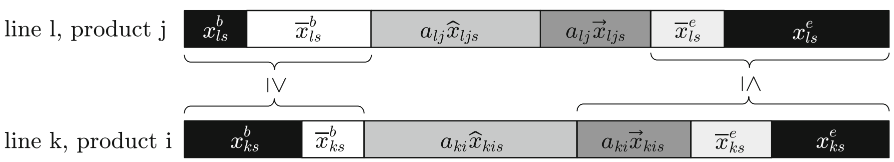

Fig. 4 Production of successor product j on line l must neither start nor end before predecessor product i on line k in every microperiod s.

4 Solution approaches

The purpose of Meyr (2004) and the above section was to propose a model formulation that represents multi-stage production systems of consumer goods industries as aggregately as possible, yet as accurately as necessary from a real-world problem's point of view. However, they have not provided an analysis of the solution performance. Of course, it does not seem to be realistic to solve GLSPMS instances of practical size to optimality by using standard MIP solvers as the feasibility problem for the GLSPMS is already NP-complete for $m_{lj} > 0$ and the special case of a single production line/stage with $e_j^{\max} = g^{\max} = 0$ (Fleischmann and Meyr 1997). Note that by using the model relaxation (increasing overtime $g^{\max}$ and external purchase $e_j^{\max}$ so that they are not restrictive any more) a first solution can be constructed quite easily. Thus the complexity of finding a feasible solution is reduced, yet the problem of finding an optimal solution becomes even more difficult because the solution space increases.

Modern solution heuristics like relax-and-fix use MIP solvers for subproblems when trying to find ("construct") first feasible solutions that may be improved afterwards by a neighborhood search, for example. Thus it seems worth checking whether reformulation techniques can strengthen the above model formulation of the GLSPMS and improve the solution performance of standard MIP solvers. Therefore, in Sect. 4.1 some well-known reformulation techniques, which have been proven successful to tighten other dynamic lot-sizing problems, are adapted to the GLSPMS. For demonstration purposes some (rather simple) construction heuristics are proposed in Sect. 4.2 to allow for a comprehensive testing of the reformulations and their potential application when using the MIP-solver Xpress MP (Sect. 5).

4.1 Reformulations

We consider some reformulation techniques which are based on variable redefinitions. We refer to them as "extended formulations" in the following (Pochet and Wolsey 2006, p. 191). Furthermore, with the help of valid inequalities (Sect. 4.1.2) the solution space can be additionally reduced, also resulting in tighter model formulations.

4.1.1 Extended formulations

Jans and Degraeve (2007) assert that typically three kinds of reformulations are used for dynamic lot-sizing problems. A common way is the variable redefinition presented by Eppen and Martin (1987). They regard production quantities as proportion of accumulated demands. The resulting formulation equals a shortest route (SR) problem.

Another approach is based on the idea that an item/period combination can be interpreted as a "facility location" (see Krarup and Bilde 1997; Rosling 1986). The opening of such a facility/plant causes fixed costs, which is similar to setting up a machine for production. Furthermore, the demands of subsequent periods can be interpreted as the demand of customers and thus transportation costs equal inventory holding costs. Hence, it represents a simple plant location (SPL) problem.

The multicommodity-flow formulation proposed by Pochet and Wolsey (1994) is quite similar to SPL. Now, the variables represent the production of an item that is used for the demand of a certain final item in a certain period.

As Jans and Degraeve (2007) state these three formulations produce the same lower bound for the single level capacitated lot-sizing problem and can be transformed to each other, respectively. Moreover, Denizel et al. (2008) have proven that SPL and SR give the same LP bound for the CLSP with setup times. Since the same observation is made by Stadtler for the multi-level CLSP (Stadtler 1996), we only concentrate on the SPL formulation presented by Stadtler as one representative. Because its extension to quantity splitting and external purchasing is not straightforward, we summarize it in the following.

For that purpose the net demand $d^n_{j\tau}$ of product j in microperiod τ, which describes the actual quantities to produce, has to be initialized first. It is calculated recursively by $d^n_{j\tau} := D^n_{j\tau} - D^n_{j,\tau-1}$ for all $\tau = 1, \ldots, S$, where $D^n_{j\tau} := \max\{0; \sum_{\sigma=1}^{\tau}(d_{j\sigma} + \sum_{i\in\mathcal{N}_j^I} p_{ji} d^n_{i\sigma}) - I_{j0}\}$ represents the cumulated (primary and secondary) net demand of product j until microperiod τ. Note that $D^n_{j0} := 0$, $d^n_{i\tau} := 0$ for all $i \in \mathcal{N}_j^I$ with $j = 1, \ldots, E$ being final items, and that due to (7) an ending inventory I_{jS} has to be built up, which increases the original primary demand d_{jS} of the last microperiod S by I_{j0} (thus leading to a re-definition of d_{jS} according to $d_{jS} := d_{jS} + I_{j0}$). The further notation and the adapted constraints are as follows:

Indices

σ, τ, θ	$= 1, \ldots, S$ microperiods

Data

$d^n_{j\tau}$	Net demand of product j in microperiod τ

Variables

$q_{ljs\tau} \geq 0$	Fraction of $d^n_{j\tau}$ which is produced on line l during microperiod $s (\leq \tau)$
$\widehat{q}_{ljs\tau} \geq 0$	Share of $q_{ljs\tau}$ that can be used by successors in the same microperiod s
$\vec{q}_{ljs\tau} \geq 0$	Share of $q_{ljs\tau}$ that can (as WIP-stock) first be used by successors in the following microperiod $s+1$ so that (23) and (24) hold
$q^{\text{ext}}_{js\tau} \geq 0$	Fraction of $d^n_{j\tau}$ which is externally purchased in microperiod s

Adapted constraints

$$q_{ljs\tau} = \widehat{q}_{ljs\tau} + \vec{q}_{ljs\tau} \quad \forall l, j, s, \tau \tag{23}$$

$$\vec{q}_{lj\tau\tau} = 0 \quad \forall l, j, \tau \tag{24}$$

$$x_{ljs} = \sum_{\tau=s}^{S} d^n_{j\tau} q_{ljs\tau} \quad \forall l, j, s \tag{25}$$

$$q_{ljs\tau} \leq y_{ljs} \quad \forall l, j, s, \tau \geq s \tag{26}$$

$$\sum_{l}\sum_{s=1}^{\tau} q_{ljs\tau} + \sum_{s=1}^{\tau} q^{\text{ext}}_{js\tau} + \sum_{i\in\mathcal{N}_j} p_{ji} \sum_{s=1}^{\tau} q^{\text{ext}}_{is\tau} \frac{d^n_{i\tau}}{d^n_{j\tau}} = 1 \quad \forall j, \tau : d^n_{j\tau} > 0 \tag{27}$$

$$I_{j0} + \sum_{l} \sum_{s=1}^{\sigma} \sum_{\tau=s}^{S} (\widehat{q}_{ljs\tau} + \vec{q}_{lj,s-1,\tau}) d^n_{j\tau} + \sum_{s=1}^{\sigma} \sum_{\tau=s}^{S} q^{\text{ext}}_{js\tau} d^n_{j\tau}$$
$$\geq \sum_{\tau=1}^{\sigma} d_{j\tau} + \sum_{l,i \in \mathcal{N}^I_j} \sum_{s=1}^{\sigma} \sum_{\tau=s}^{S} p_{ji} d^n_{i\tau} q_{lis\tau} \quad \forall j > E, \sigma \tag{28}$$

The basic idea is that the variables $q_{ljs\tau}$ describe the fraction of the total (primary and secondary) net demand $d^n_{j\tau}$ of item j in period τ, which has been produced in microperiod $s (\leq \tau)$ on line l. Correspondingly, the fraction of the WIP stock for the same microperiod needs to be zero (24). So each x_{ljs} of the model presented in Sect. 3.2 has to be replaced by (25). Additionally, the linking constraints (11) are replaced by (26). Note that the "big M" is set to 1 in this case.

Since production quantities now contain the information how long they are stored, we do neither need the inventory variables I_{js} nor the inventory balancing constraints (4) any longer. Holding costs can be directly calculated using the $q_{ljs\tau}$ as it is, for example, shown in Stadtler (1996). Fractions $q^{\text{ext}}_{js\tau}$ for external purchasing also need two time indices s and τ because it might be necessary to buy and assemble a pre-product earlier than its corresponding final item is needed. Thus costs for external purchasing and constraints (18) have to be adapted analogously. Accordingly, the objective function (33) can be found in Appendix A.

Even though the inventory variables can be dropped, we still have to ensure that primary demands are fulfilled and pre-products are produced on time. Therefore, constraints (27) enforce that the net demand fractions of a product j sum up to 100 %. Here of course production quantities have to be considered, but external purchasing must not be forgotten either. This does not only concern the purchased quantities of product j itself. Purchasing direct or indirect successor products of j also reduces secondary demand for j. Thus those have also to be taken into account. However, we can restrict ourselves to products and periods with a positive net demand. Additionally, constraints (28) replace $I_{js} \geq 0$. They guarantee that cumulated supply of pre-products equals or exceeds their cumulated demand for each microperiod σ, i.e., initial inventory plus the total production or external purchase for net demand up to σ is not lower than the—up to the same period—total original gross demand and secondary net demand, which has been induced by production.

It is interesting to note that the one-period lead time of the variables $\vec{q}$ has to be taken into account ($\vec{q}_{lj0\tau} = 0$ for all l, j, τ if I_{j0} also includes the WIP-inventory) when calculating the overall production quantity of pre-product j that is available to fulfill demand of microperiod τ on the left-hand side of (28). Due to (27), the SPL formulation does not allow any final stocks. This is the reason why the final stocks (7) of the original formulation had to be included in the net demand calculation. Thus constraints (7) can be dropped now without loss of generality. In addition, note that (5) have to be adapted analogously to (28) replacing $I_{js} \geq 0$, and that variables $q_{ljs\tau}$ could be eliminated by replacing them with (23) in the respective constraints. We only introduced them to improve readability.

The above-mentioned formulations refer to the "inventory holding" and "production" variables. In contrast, Karmarkar and Schrage (1985) propose to reformulate the "setup" variables. By introducing the flow conservation constraints (29) for changeover variables $z_{lijs} \in \{0, 1\}$ and substituting y_{ljs} by $\sum_i z_{lijs}$ they eliminate constraints (15) and replace the setup state indicators y in the remaining constraints:

$$\sum_i z_{lij,s-1} = \sum_i z_{ljis} \quad \forall l, j, s > 0 \tag{29}$$

Since the SPL and the flow conservation reformulations can, but need not to be applied in combination, four different types of formulations are considered in the remainder of this paper: original (O), i.e., the base model without SPL and flow conservation presented in Sect. 3.2, flow conservation (F), simple plant location (S), and its combination (SF).

4.1.2 Valid inequalities

Besides, a formulation can be tightened by additional valid inequalities. One class of those valid inequalities is proposed by Pochet and Wolsey (2006, p. 218). They represent a subset of the known (l,S)-inequalities (see Barany et al. 1984). If there is no setup for an item for several periods $(s+1, \ldots, \tau)$, the stock in period s must equal or exceed the corresponding cumulative demand in these periods. For the GLSPMS it means that also the WIP stock in microperiod s and the purchased quantity of microperiods $s+1, \ldots, \tau$ have to be taken into account. The corresponding constraints (30) can be added to the original formulation (O) and flow conservation formulation (F), but only for the final items $(j = 1, \ldots, E)$ with primary demand.

$$I_{js} + \sum_l \vec{x}_{ljs} + \sum_{\sigma=s+1}^{\tau} o_{j\sigma} \geq \sum_{\sigma=s+1}^{\tau} d_{j\sigma} \cdot \left(1 - \sum_{l,i} \sum_{\vartheta=s+1}^{\sigma} z_{lij\vartheta}\right)$$
$$\forall j \leq E, s < S, \tau > s : d_{j\tau} > 0 \tag{30}$$

Note that these inequalities can be adapted and limited to (inventories at the end of) macroperiods t because positive demand can only occur during the last microperiod of a macroperiod, i.e., $d_{js} = 0$ for $s \notin \Lambda$ (see Sect. 3.2).

In contrast, since the SPL formulation is based on echelon stocks (Clark and Scarf 1960), which represent the system-wide stocks of an item (on hand or already built in successor items), in the SPL-based formulations the (l,S)-inequalities can be applied for all products. Accordingly, constraints (31) enforce that if there is no setup, all fractions that are produced or externally purchased [directly or already built in externally purchased successor items, cp. (27)] must sum up to 1.

$$\sum_{\vartheta=1}^{s} \sum_{\sigma=s+1:d^n_{j\sigma}>0}^{\tau} \left(\sum_l q_{lj\vartheta\sigma} + q^{\text{ext}}_{j\vartheta\sigma} + \sum_{i \in \mathcal{N}_j} p_{ji} \frac{d^n_{i\sigma}}{d^n_{j\sigma}} q^{\text{ext}}_{i\vartheta\sigma} \right)$$
$$\geq \sum_{\sigma=s+1}^{\tau} \left(1 - \sum_{l,i} \sum_{\vartheta=s+1}^{\sigma} z_{lij\vartheta} \right) \quad \forall j, s < S, \tau > s : d^n_{j\tau} > 0 \tag{31}$$

Moreover, as mentioned above lot-sizing models usually need a setup forcing constraint like (11). Often a good choice of the "big M" increases notably the lower bound of the LP relaxation and thus reduces computational runtime. Since this value represents an upper bound for the production quantity, a widespread approach is to define M as the minimum of accumulated remaining demand and available capacity in the current period divided by the production coefficient. Note that new index sets S_t are introduced, which contain all microperiods belonging to a macroperiod t:

$$x_{ljs} \leq \min \left\{ \frac{K_t}{a_{lj}}, \sum_{\sigma=s}^{S} d^n_{j\sigma} \right\} \cdot \sum_i z_{lijs} \quad \forall l, j, t, s \in \mathrm{S_t}. \tag{32}$$

In the remainder of this paper (K) denotes the stock-inequalities (30) for the formulations (O) and (F) and the echelon stock inequalities (31) for the formulations (S) and (SF), respectively. Furthermore, (M) denotes the valid inequalities (32) for the big-M variant.

4.2 Construction heuristics

In Sect. 4.1 different formulations are presented that can improve the LP lower bound of the GLSPMS. When used within a standard MIP solver they might give solutions that are nearly integral and allow finding optimal solutions substantially faster than using the original formulation of Sect. 3.2. But if the exact optimization still proves to be too slow, heuristics are needed to find good feasible solutions in a reasonable time (Pochet and Wolsey 2006, p. 108).

The simplest heuristic approach—that will also serve as a benchmark in the computational tests of Sect. 5—is to run the standard MIP solver for a fixed amount of time and to take the best solution found so far. This approach will be called "Truncated MIP" (TM) in the following. For our tests we will limit the runtime to 300 s.

The idea of the second heuristic is to fix those binary variables that are already integral in the solution of the LP relaxation. Afterwards the remaining MIP is tried to be solved (Pochet and Wolsey 2006, p. 108). If only a few variables of this "LP-and-Fix" (LF) heuristic get fixed in the first step, the computation times of the remaining MIP can still become crucial. Thus, runtime is limited to 300 s as well.

The "Relax-and-Fix" (RF) heuristic partitions the binary variables into several subsets. For each subset an MIP will be solved, where only those variables which belong to this corresponding subset are restricted to binary values. All others are relaxed. These MIPs are solved sequentially, i.e., after an MIP is solved, its binary variables are fixed and the next subset is considered (Pochet and Wolsey 2006, p. 109). To build the subsets of variables we divide the planning horizon into time windows. Each time window consists of a macroperiod and its corresponding microperiods. The time windows are rolled from macroperiod 1 to the end of the planning horizon. In order to be able to compare all heuristics in a fair way, the total runtime for the Relax-and-Fix must also not exceed 300 s. Thus, each subproblem, which needs to be solved, must not exceed $300/T$ s where T is the number of macroperiods.

5 Computational tests

As shown by Meyr (2004) and sketched in Sect. 3 the GLSPMS appears to be appropriate to model practical lot-sizing and scheduling problems of multi-stage flow shop production in consumer goods industries with a sufficient level of detail. Nevertheless, it has to be tested how the GLSPMS behaves from a computational point of view—when either solved exactly by a standard MIP solver or heuristically by one of the MIP-based solution approaches presented.

For this purpose several test scenarios were created, which are described in the next Sect. 5.1. All formulations and heuristic approaches have been implemented and tested with Xpress MP of Fair Isaac. Accordingly, Xpress Optimizer Version 20.00.22 was used as a standard MIP-solver. The test instances, which will be described in the following, were run on a single core (2 threads) of a Dell Precision T1500 with an Intel Core i7-860 (2.8 GHz) processor, 8 GB RAM, and the Windows Server 2008 64bit operating system.

5.1 Base scenarios

We manually created three base scenarios with a divergent, general, and serial product structure. They are motivated by practical problems and represent realistic situations. Even though the dimensions may differ from reality, the data allow for comparing different formulations and solution approaches. In the following the structure of these scenarios is illustrated. The complete data can be found in Appendix B.

5.1.1 Serial product structure (SER)

The first scenario is a three-level serial production of juices. There are two final items, a six-pack of apple juice ($j = 1$) and a six-pack of cherry juice ($j = 2$). The apple juice is filled in standard 1L bottles ($j = 3$), whereas the other, more outstanding 1L bottles—differing in shape and texture for marketing purposes—are used for the cherry juice ($j = 4$). Accordingly, in a first step, two different types of PET bottles ($j = 5, 6$) are produced on stretch blow-molding machines. Since filling is faster than molding, two identical molding lines $l = 3, 4$ are needed. A schematic representation of the corresponding BOM and the assignment of products to lines is given in Fig. 5.

On the second production stage (and level $e = 1$ of the BOM structure, respectively) the bottles are filled with the help of a single tank $l = 2$. Since this tank has a minimum fill level, minimum lot-sizes are required. Setup times are highly sequence dependent

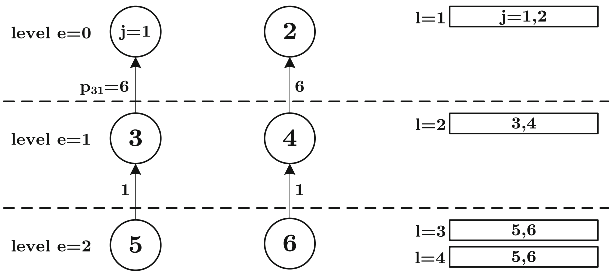

Fig. 5 BOM and product-line assignment of the serial instance

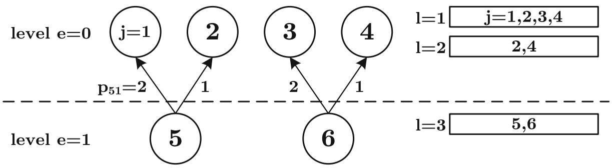

Fig. 6 BOM and product-line assignment of the divergent instance

due to cleansing processes. As mentioned, filling is quite fast, however, WIP-stock of filled bottles is limited because of the restricted space between the filling and the wrapping machine.

On the final stage ($l = 1$) the shrink-packaging takes place, where six bottles of each sort are wrapped. Due to the different types of bottles the corresponding calibration is quite time-consuming. Each macroperiod has a capacity of 80 time units and the demand varies between 2 and 8 units for the final items. The production costs are the same (=1) for all lines, except for line 4, which causes higher costs for maintenance reasons. The lines are already setup for products 1, 3, 5, and 6, respectively. The remaining data can be found in Table 7a–n of Appendix B.

5.1.2 *Divergent product structure (DIV)*

In this divergent scenario sheet glass is produced in different colors and lengths on two production stages. The scenario is illustrated in Fig. 6. On the first stage there is a single line $l = 3$ producing light-colored ($j = 5$) and dark-colored ($j = 6$) glass. The production of light-colored glass is more time-consuming ($a_{35} = 4$) than the production of the dark one ($a_{36} = 2$). To change the setup from light to dark is less complex ($st_{356} = 2$) than vice versa ($st_{365} = 6$). Both pre-products can be cut in two lengths—short and long—on the second stage. Thus, there are four final items $j = 1, \ldots, 4$ possible. Long items need twice as much of glass ($p_{51} = p_{63} = 2$) than short cut ($p_{52} = p_{64} = 1$).

For cutting, two parallel lines are available. The newer one $l = 1$ needs $a_{11} = a_{13} = 3$ for a long and $a_{12} = a_{14} = 4$ for a short item. Changeover from short to long takes 2 time units, whereas long to short only 1. Besides, a cleansing of 3 time units is needed for a change from dark to light and of 1 time unit for the reverse. The older line $l = 2$ is just able to produce short items in 8 units of time. On that machine, a

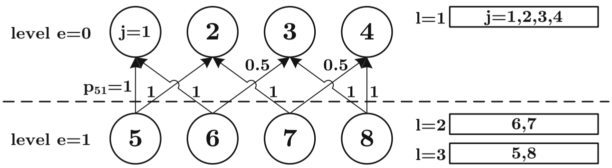

Fig. 7 BOM and product-line assignment of the instance with the general product structure

setup change from dark to light requires 4 and 2 for the reverse, respectively. Stocks are not allowed for the pre-products.

The minimum lot-size is one unit. At the beginning all production lines $l = 1, 2, 3$ are already set up for $j = 1, 2, 6$. The older one causes production costs of 2 per unit, the other lines expense 1 per unit. Setup costs equal the corresponding setup times. The planning horizon consists of three macroperiods $t = 1, \ldots, 3$. Accordingly, there are four microperiods with fixed starting times of 0, 80, 160, and 240, the last one being the artificial one. There is only demand for final items varying between 2 and 6 units. All other values are listed in Table 8a–n of Appendix B.

5.1.3 General product structure (GEN)

A third scenario with a general product structure describes the production of yogurt. Two kinds of yogurt ($j = 6$ and $j = 7$) are filled in big and small cups ($j = 5$ and $j = 8$) resulting in four different final items $j = 1, \ldots, 4$. In this make-and-pack environment, the cups as well as the yogurt are produced on the first stage, each of them on a dedicated production line, which is shown in Fig. 7. The cups on line $l = 3$ need $a_{35} = 8$ and $a_{38} = 2$ time units per item. A changeover st_{358} from the big to the small packaging format needs 8 units of time and 6 for the reverse. Besides, after a machine breakdown ($i = 0$) it takes $st_{30j} = 12$ to ramp up. But standby is also possible with $b_3 = 1$. The yogurt can be produced on line $l = 2$ in parallel with $a_{2j} = 6$. Both setups are sequence independent ($st_{267} = st_{276} = 4$).

On the second stage the final items are filled and packed on line $l = 1$ using $p_{63} = p_{74} = 0.5$ units of yogurt for small cups, else 1. Making the yogurt on line $l = 2$ and filling it on line $l = 1$ are synchronized. Thus, stocking for items 6 and 7 is not allowed. However, producing the cups is decoupled by an unlimited buffer. The corresponding production coefficients are $a_{11} = a_{12} = 6$ and $a_{13} = a_{14} = 3$. A changeover from big to small cups needs 8 and from small to big 6 time units, whereas changing only the yogurt flavor needs 4. All further data is presented in Appendix B (Table 9a–n).

As shown in Table 1, these three base scenarios are varied to 82 problem instances by copying BOM structures and increasing the number of lines and periods of demand. Finally, instances with 6–12 products, 3–9 lines, and 3–8 macroperiods result.

5.2 Results

At first, the aim is to learn the advantages and disadvantages of the different model formulations and valid inequalities presented in Sect. 4.1. For the best performing

Table 1 Overview of test instances

	L	J	T
SER	4,8	6,9,12	4,5,6,7,8
DIV	3,6,9	6,9,12	3,4,5,6
GEN	3,6	8,11	3,4,5,6

ones, the influence of different problem characteristics like product structure, utilization rate, etc., is examined in Sect. 5.2.2. Finally, the solution quality of the heuristics of Sect. 4.2 is compared in Sect. 5.2.3.

5.2.1 Comparison of the different model formulations

Before the different formulations were tested, the maximum solvable problem size and the impact of different solver settings like "automatic cut generation" and "presolving" of the standard MIP solver have been analyzed. These pre-tests have shown that the default solver settings perform best on average. They are thus used for all experiments of Sect. 5.2.

In the following the extended formulations "flow conservation" (F), "simple plant location" (S) and their combination (SF) of Sect. 4.1.1 are compared with the original formulation (O), each with the stock inequalities (K) and valid inequalities for the big-M variant (M) of Sect. 4.1.2 potentially being added. According to the observations of the pre-tests, we differentiate the test instances by their size and use default solver settings in the following. The "size" of a problem instance is defined as the number of products multiplied by the number of lines and the total number of microperiods ($J \times L \times S$). All in all, three groups of sizes result, which are called *G1, G2* and *G3*, where *G1* contains all instances with $J \times L \times S \leq 250$, *G2* with $250 < J \times L \times S \leq 500$, and *G3* with $J \times L \times S > 500$. The results are shown in Table 2.

The first column (PS) describes the percentage of instances that are solved in 3,600 s to optimality. For those instances the percentage integrality gap $= \frac{\text{optimal solution} - \text{LP relaxation}}{\text{LP relaxation}} \times 100$ can be measured, which is given in the second column (IG). The quality of the different formulations is evaluated by inspecting the IG, since the LP relaxation describes the tightness of the formulation: the smaller the gap within the same group of instances, the better the formulation. In the third column the average runtime (AR) of the solved instances is presented (s).

It is interesting to note that using the simple plant location formulation (S, SF) produces better IG values (36 and 45 % on average) than are possible when using the original formulation or F stand alone (56 and 50 % on average). These observations coincide with the results of Denizel and Sural who regarded the CLSP with setup times (Denizel and Süral 2006).

If S is not applied, the IG can at least be improved by adding the big-M inequalities M (e.g., for O and F the IG can be reduced from 80 to 54 % and from 76 to 46 % when not using K, and from 49/44 % to 42/35 % when using K). Unfortunately however, they do no make it possible to solve more instances. The percentage of instances PS

Table 2 Percentage of test instances solved to optimality in 3,600 s (PS), the corresponding integrality gap (IG), and the average runtime (AR) in seconds for the original formulation (O), the extended formulations "flow conservation" (F), "simple plant location" (S), and their combination (SF)—possibly complemented by stock inequalities (K) and big-M-inequalities (M)

	–			M			K			M + K			Average		
	PS	IG	AR	PS	IG	AR	PS	IG	AR	PS	IG	AR	PS	IG	AR
G1															
O	100	126	13	100	73	13	100	69	11	100	54	12	100	80	12
F	100	127	19	100	65	11	100	67	18	100	50	25	100	77	18
S	100	41	13	100	50	9	100	41	11	100	46	12	100	44	11
SF	100	71	15	100	65	14	100	58	14	100	51	14	100	61	14
G2															
O	72	54	477	72	49	417	67	41	545	74	40	558	71	46	499
F	71	49	424	75	42	430	68	36	428	73	34	468	72	40	438
S	72	34	415	73	38	371	70	33	448	75	37	506	73	36	435
SF	70	44	480	74	45	463	68	40	412	75	39	506	72	42	466
G3															
O	22	4	349	27	7	634	25	4	603	27	7	568	**25**	6	546
F	30	4	451	37	4	732	38	6	927	44	6	860	**37**	5	763
S	27	3	976	29	4	513	25	3	709	25	3	487	27	3	671
SF	30	4	414	33	6	565	35	6	678	35	4	597	33	5	569
Average															
O	66	**80**	269	68	**54**	273	65	**49**	322	69	**42**	336	**67**	**56**	300
F	68	**76**	261	72	**46**	308	69	**44**	336	73	**35**	363	**70**	**50**	317
S	68	**34**	309	69	**39**	239	66	33	291	69	37	302	**68**	**36**	285
SF	68	50	280	70	48	297	68	43	285	71	39	324	**69**	**45**	297

Table 3 Percentage of instances that are unfinished after 3,600 s (PU) and the corresponding duality gap (DG) for the original formulation (O), the extended formulations “flow conservation” (F), “simple plant location” (S), and their combination (SF)—possibly extended by stock inequalities (K) and big-M-inequalities (M)

	–		M		K		M + K		Average	
	PU	DG	PU	DG	PU	DG	PU	DG	PU	DG
G2										
O	28	31	28	27	33	25	26	28	**29**	**28**
F	29	30	25	24	32	23	27	20	28	24
S	28	26	27	24	30	25	25	25	**27**	25
SF	30	24	26	18	32	21	25	20	28	**21**
G3										
O	78	15	73	17	75	17	73	14	75	16
F	70	18	63	18	62	18	56	18	63	**18**
S	73	14	71	13	75	14	75	13	73	**14**
SF	70	16	67	15	65	17	65	15	67	16
Average										
O	34	21	32	21	35	20	31	19	33	20
F	32	22	28	20	31	20	27	19	30	21
S	32	19	31	18	34	18	31	18	32	18
SF	32	19	30	16	32	18	29	17	31	18

that can be solved to optimality within the time limit almost stays the same regardless of the type of formulation used (67–70 % on average).

What can also be seen is that for the simple plant location formulation (S) the big-M inequalities produce higher integrality gaps (from 34 to 39 %). The plain explanation is that constraints (26) are tighter than the corresponding adapted constraints (32).

As was to be expected, less instances can be solved to optimality when the problem sizes increase. Within group G1 all problem instances can be solved within the one hour time limit (PS = 100). This reduces to an average of 25–37 % within group G3. Within the groups G1 and G2 the values of PS are quite independent on the formulation chosen. However, for the big instances within G3 the flow formulation F performs significantly better than the others (e.g., 37 % compared to 25 % of the original formulation). But obviously, only those G3 instances which show a small IG can be solved to optimality at all.

Concerning valid inequalities it can be stated that it is always advantageous to use the big-M-inequalities. The PS-values of column “M” are never worse than the corresponding ones of column “–”. The same holds true when comparing columns “M + K” and “K”. All in all the combination M + K performs best with respect to the number of optimally solvable instances PS.

To complete the picture, Table 3 also shows performance indicators for the problem instances that have proven to be too difficult to be solved to optimality within the time limit. The first column PU—showing the percentage number of unfinished problems—complements the PS column of Table 2. Correspondingly, group G1 is dropped since all instances of this group have been solved to optimality. For all other instances, the percentage duality gap (DG) after 3,600 s is shown. It is measured by: duality gap $= \frac{\text{best solution} - \text{best lower bound}}{\text{best lower bound}} \times 100$. It gives an idea about the worst case quality of the best (feasible) solution found after 1 h.

Interestingly, the duality gaps DG decrease when the problem sizes increase from G2 (21–28 %) to G3 (14–18 %). This appears counterintuitive at the first sight. But one has to keep in mind that—because of their smaller size—within the 1 h time limit more than twice the number of G2-instances have been solved to optimality as compared to G3-instances. Thus after 1 h only the 27–29 % "hard" instances of the middle-size class G2 remain, which have a quite high DG. The larger instances of class G3 need longer computation times anyway. Thus, after one hour also some "easier" problem instances are left which already show a rather low DG.

It is also interesting to note that the S formulations offer preferable duality gaps (the DGs of S are never higher than the ones of O; the same holds true for SF and F), even though the percentages of unfinished instances PU might be higher (F and SF for G3).

5.2.2 Influence of varying problem characteristics

So far the instances have only been examined in terms of their size. But it might also be important to learn about the impact of the underlying product structure or other characteristics of the test instances like the utilization rate, the time between orders and the relation between setup costs and setup times.

For this reason Table 4 shows the same results as Table 2, but now grouped by the product structure. Indeed, three main effects can be seen: first, the big-M variants M and M + K are still preferable (best value per model formulation is marked in italics). Usually, M + K performs best, but especially for divergent product structures M also shows very good results. Second, the integrality gap of the serial instances is very low compared to the other product structures (between 5 and 9 % on average). In fact, almost all instances of group G3, which have been solved to optimality according to Table 2, actually show a serial product structure. Thus, divergent and general problem structures seem to be significantly more difficult than serial ones. And third, the flow formulation is beneficial for serial and general product structures. The PS values of F outperform O and the PS values of SF outperform S for these kinds of problems. In most cases F again seems preferable to SF. However, the situation is different for the divergent instances. Here, F seems to be rather counterproductive and S performs best (followed by O).

Summing up, the flow formulation F in combination with both valid inequalities M + K performs best for the serial instances (88 %) and for the general instances (78 %). On the contrary, F is rather disadvantageous for divergent product structures. Here, the simple plant location formulation S—again using the extension of M + K—is superior (66 %).

In accordance with these results, with the help of the two "best" formulations F + M + K and S + M + K we want to examine the potential influence of the other problem characteristics mentioned. To see the impact of the utilization rate, the capac-

Table 4 Percentage of test instances solved to optimality in 3,600 s (PS), the corresponding integrality gap (IG), and the average runtime (AR) in seconds for all formulations—now grouped by the product structure

	–			M			K			M + K			Average		
	PS	IG	AR	PS	IG	AR	PS	IG	AR	PS	IG	AR	PS	IG	AR
SER															
O	70	11	176	73	9	222	73	7	286	*74*	6	205	72	**9**	223
F	78	11	192	84	8	291	83	7	352	***88***	6	335	83	8	295
S	74	5	373	*75*	6	165	73	5	324	74	6	258	74	**5**	279
SF	78	9	250	79	8	187	79	7	275	*83*	7	306	79	8	255
DIV															
O	63	124	375	*64*	95	352	59	92	388	*64*	75	480	62	96	399
F	60	121	409	*62*	85	385	56	87	307	59	69	348	59	91	363
S	64	58	342	65	66	345	62	57	339	***66***	64	398	64	61	356
SF	58	92	327	*63*	87	439	58	79	325	61	73	337	60	83	358
GEN															
O	68	110	223	68	54	208	66	44	263	*70*	41	289	68	62	246
F	70	111	123	72	48	202	74	46	352	***78***	37	436	74	59	283
S	*66*	37	131	*66*	46	163	*66*	37	140	*66*	39	185	66	40	154
SF	72	54	253	72	48	237	72	45	239	*74*	41	335	73	47	267
Average															
O	66	80	269	68	54	273	65	49	322	69	42	336	67	56	300
F	68	76	261	72	46	308	69	44	336	73	35	363	70	50	317
S	68	34	309	69	39	239	66	33	291	69	37	302	68	36	285
SF	68	50	280	70	48	297	68	43	285	71	39	324	69	45	297

Table 5 Percentage of test instances with different problem characteristics solved to optimality in 3,600 s (PS), the corresponding integrality gap (IG), and the average runtime (AR) in seconds for the two best formulations S and F including stock inequalities (K) as well as big-M-inequalities (M)

	Normal			Cap			tbo			Setup		
	PS	IG	AR	PS	IG	AR	PS	IG	AR	PS	IG	AR
SER												
F + M + K	88	6	335	89	5	129	89	7	195	88	6	302
S + M + K	74	6	258	73	5	125	76	7	326	75	6	340
DIV												
F + M + K	59	69	348	**57**	**39**	461	66	68	416	63	70	347
S + M + K	66	64	398	**55**	**40**	226	65	70	326	64	69	365
GEN												
F + M + K	78	37	436	76	**70**	320	78	50	405	82	39	464
S + M + K	66	39	185	72	**79**	446	66	56	280	66	39	124
Average												
F + M + K	73	35	363	72	32	288	76	40	325	75	37	356
S + M + K	69	37	302	65	36	240	69	43	317	68	38	305

ity (cap) of the instances is reduced by 20 % ($\bar{w}_s^{\text{cap}} := 0.8\bar{w}_s \; \forall s \in \Phi$). The time-between-orders (tbo) is supposed to be a second factor. Therefore, the setup costs are doubled ($s_{lij}^{\text{tbo}} := 2s_{lij} \; \forall l, i, j$) so that holding inventory becomes more attractive and larger lot-sizes are to be expected. Finally, since setup costs and times are the same for the current instances, the setup costs are now "inverted" (setup). This means that for each line the setup cost for the most expensive changeover is now the cheapest, the second most expensive is now the second cheapest, and so on ($s_{lij}^{\text{setup}} := \max_{i,j}\{s_{lij}\} + \min_{i,j}\{s_{lij}\} - s_{lij} \; \forall l, i, j \neq i$ with $a_{li} > 0$ and $a_{lj} > 0$). Thus, setup costs and setup times are not proportional any longer.

In Table 5 the corresponding results are shown. In column "normal" the values of Table 4 are repeated for comparison. Obviously, general conclusions in terms of solvability can hardly be derived. For instance, by reducing the capacity the IGs for the divergent product structures become smaller (39/40 %), but less instances could be solved (57/55 %). In contrast, PS is nearly the same for the general structure with the flow formulation and even higher for the simple plant location formulation, although the IGs are substantially larger now (70/79 %).

The impact of a different time-between-orders is almost negligible in terms of PS. Only the IGs increase slightly for most of the instances, again to the highest degree for the general product structures. The reason is that—compared to the other cost terms of the objective function—mainly the setup costs are "affected" by the LP relaxation. By allowing fractional setup costs instead of binary ones, the influence of an increase gets mitigated.

For the "setup" instances no real difference could be observed at all. Accordingly, it seems that instances with proportional setup times and costs are *not* easier to solve.

5.2.3 Heuristics

In the remainder we evaluate the solution quality of the heuristics described in Sect. 4.2. Table 6 contains the results of the Truncated MIP (TM), the LP-and-Fix (LF) and the Relax-and-Fix (RF), again for each possible formulation and with a time limit of 300 s.

In the columns denoted by "GP" the average percentage difference between the best feasible solution found within 300 s and the overall best lower bound found by any formulation within 3,600 s (results of Sect. 5.2.1) is given. Thus, differences really stem from the formulations' potentials to find feasible solutions. Moreover, the average runtime (AR) is listed as well (s).

Once more it can be seen that the serial instances are easier to solve. TM as well as RF are both able to solve most of the instances to optimality (all GPs $\leq$ 2 %)—independently from the used model formulation. Only LF shows some weaknesses when using the flow formulations. Then GP values up to 23 % occur. But with the other formulations LF produces competitive results as well, even though the average runtimes are significantly higher than those of TM and RF. Note that RF also runs considerably faster than TM.

For the divergent instances all three heuristics perform quite differently. TM performs best (GP between 7 and 16 %) and specially the S formulation shows again its superiority (GP between 7 and 9 %) as in the section before. In contrast, RF is not that good anymore. The gaps (GP) lie between 19 and 29 %, but with an average runtime between 11 and 20 s as compared to the 14–28 s of TM. Finally, the LF produces the worst results. The lowest gap is 31 %, the highest even 663 %. Again a systematic difference related to the formulations can be observed. It seems that for the F and SF formulations more variables can be fixed. On the one hand, this results in a lower runtime. But on the other hand, the GP values are quite high.

This contrast between F and SF for LF becomes dramatically obvious for the instances with a general product structure. Here, LF produces e.g., a gap of 1,108 % in 1 s compared to e.g., a gap of 25 % after an average runtime of 109 s. The results of TM are again the best with gaps between 8 and 20 %. SF shows the best quality, but with the longest running times. The quality of F is close by, but computation times are advantageous. RF performs quite bad (GP up to 213 %). However, the M+K inequalities allow competitive results for the O, F, and S formulations (29–36 %).

In sum, TM performs best by far in terms of solution quality, in particular the S formulation for the DIV instances and the (F and) SF formulation(s) for the GEN instances. The main reason is probably that these formulations benefit from their similar behavior in Table 4. LF is fast, but of very bad quality when applied in a flow formulation. The O and S formulations of LF behave substantially better with respect to solution quality, however, at costs of higher computation times. RF is also worse than TM in terms of solution quality. However, the computational effort for RF is significantly lower. This might become interesting when real-world instances are considered, which usually are much larger than the rather small examples tested here. In this case, running times, scalability, and thus also decomposability of a heuristic approach will become crucial.

Note that an increase of computation time from 300 to 3,600 s does not really change the picture. For TM the GP stays almost the same when the SER instances

Table 6 Gap (GP) and average runtime (AR) of Truncated MIP (TM), LP-and-Fix (LF), and Relax-and-Fix (RF) for the original formulation (O), the extended formulations "flow conservation" (F), "simple plant location" (S), and their combination (SF)—possibly extended by stock inequalities (K) and big-M-inequalities (M)

	–						M					
	TM		LF		RF		TM		LF		RF	
	GP	AR	GP	AR	GP	AR	GP	AR	GP	AR	GP	AR
SER												
O	1	14	1	118	2	7	1	14	1	114	2	8
F	1	15	23	3	2	5	1	13	22	3	1	6
S	1	19	1	113	1	10	1	18	1	96	1	10
SF	1	25	20	5	2	8	1	21	19	4	1	9
DIV												
O	10	20	66	104	29	11	10	19	62	99	26	12
F	16	16	663	2	28	14	13	15	632	2	24	14
S	7	19	31	102	23	14	7	18	42	101	21	12
SF	13	28	637	2	26	15	10	26	644	2	20	14
GEN												
O	18	21	42	99	213	5	19	17	25	99	87	5
F	15	18	1,119	1	193	5	13	15	**1,108**	**1**	84	6
S	19	17	20	99	39	8	18	16	**25**	**109**	48	8
SF	10	28	1,082	1	207	5	9	24	1,093	1	85	7
	K						M + K					
	TM		LF		RF		TM		LF		RF	
	GP	AR	GP	AR	GP	AR	GP	AR	GP	AR	GP	AR
SER												
O	1	16	1	126	2	9	1	13	1	113	1	13
F	1	15	23	3	1	6	1	13	21	3	1	6
S	1	18	1	123	1	14	1	16	1	109	1	14
SF	1	23	19	5	1	11	1	19	18	5	1	13
DIV												
O	12	19	46	119	24	16	8	18	43	103	22	15
F	16	15	543	2	27	20	12	14	519	2	26	19
S	9	19	33	108	22	14	6	19	36	108	19	16
SF	13	25	589	3	26	17	10	25	583	3	23	15
GEN												
O	16	17	14	109	42	7	17	17	16	97	36	8
F	9	17	1, 057	1	41	6	8	15	1, 026	1	29	6
S	16	17	16	121	54	10	20	17	20	109	35	9
SF	9	26	1, 065	1	146	6	9	24	1, 048	2	66	7

are considered. The reductions ($GP^{300\ s} - GP^{3,600\ s}$) range between 1 and 5 for the DIV instances with F+M+K being the only exception because there a reduction by 11 is possible. The GEN instances are reduced by 1–6 as compared to Table 6. The results of LF and RF cannot noticeably change when longer running times are allowed. Within 300 s, all remaining MIPs of the (bad performing) flow formulations of LF and around 70 % of the other formulations have already been solved to optimality. Thus there is not much room left for improvement. For RF more than 99.5 % of the "sub-MIPs" of RF can be solved to optimality within 300 s. Here improvements would only be possible if the sub-MIPs were enlarged, i.e., by defining larger time windows. However, testing this would go beyond the scope of this paper.

6 Conclusions

An improved version of the "General Lot-sizing and Scheduling Problem for Multiple production Stages" (GLSPMS), a model for simultaneous multi-item, multi-level lot-sizing and scheduling, is presented. In this model deterministic, dynamic demand has to be met without backlogging. Sequence dependent setup times may further reduce the limited capacity of heterogeneous, parallel production lines per stage. The objective is to minimize the sum of production costs, sequence-dependent setup costs, and holding costs as well as costs for external purchase, overtime, and standby.

The main idea of this model is to use a common time grid for all production lines of the different stages, which is based on rather short (micro-) periods, whose starting times are decision variables. This enables us to model realistically short lead times between the different stages without generating an unnecessarily high number of periods.

Different reformulations of this model, which aim at improving its computational performance without destroying its basic structure, have been tested using a standard MIP solver. These reformulations, which had already proven to be successful for other types of lot-sizing problems, needed to be adapted for the problem on hand. As shown in Sect. 5.2.2, the "simple plant location" based reformulation appears to perform slightly better for instances with a divergent product structure, whereas a flow formulation shows advantages for all other instances. Furthermore, the usage of a subset of the (l,S) valid inequalities as well as tightened setup forcing constraints are always recommendable. Nevertheless, even for instances of moderate size the percentage of unfinished problems (that cannot be solved to optimality within a reasonable amount of time) as well as the corresponding duality gaps are quite high.

Thus, standard MIP solvers do not appear to be promising for solving this problem satisfactorily. For this reason, some first solution heuristics have been designed and tested as well. Here Truncated-MIP performed best in terms of solution quality as compared to LP-and-Fix and Relax-and-Fix. However, it has to be kept in mind that only problem instances of moderate size have been considered. Real-world problems are of larger size and thus enforce more powerful, scalable solution approaches to be developed. Then especially the runtime performance of a Relax-and-Fix approach might be advantageous.

All in all, the presented results promise to show further value in the future. For example, the best-performing TM and Relax-and-Fix variants can be used to construct initial solutions, which might be improved in a second step by meta-heuristics like local search and evolutionary algorithms. Alternatively, other math-programming-based heuristics might be developed, which take advantage of the reformulations and valid inequalities tested. Such more sophisticated solution approaches should be a topic of future research.

Acknowledgments We want to thank the three unknown referees and Laurence Wolsey for their helpful feedback and their valuable suggestions for improvement.

Appendix A: Objective function of the SPL formulation

To apply the SPL approach (Krarup and Bilde 1997) for multiple production stages, Stadtler (1996) uses echelon stocks. This means that for each item j the system-wide stock level is determined considering the stocks on hand as well as the stocks already built in direct or indirect successor products. Since the echelon stocks do not indicate on which production stage the regarded item is located, marginal holding costs h^{m}_{js} have to be introduced. They describe the delta of the holding costs h_{js} to the summed holding costs of the already processed pre-items (see e.g. Stadtler 1996 for a formal definition). For each echelon these marginal holding costs must be added to the total costs. In addition to the notation used before, the following indices and data are necessary:

Indices

ρ, u $\quad = 1, \ldots, T$ macroperiods

Index set

S_t $\quad$ set of all microperiods belonging to macroperiod t

Data

h^{m}_{js} $\quad$ marginal holding costs of product j in microperiod s

$$
\begin{aligned}
\text{Min} & \sum_j \sum_{t=1}^{T-1} \sum_{u=t+1}^{T} \sum_{\rho=t}^{u-1} h^{\mathrm{m}}_{j,f_{\rho+1}-1} \sum_{s \in S_t} \sum_{v \in S_u} d^n_{jv} \sum_l q_{ljsv} \\
& + \sum_j \sum_{s \in \Lambda} h_{js} \cdot \max\left\{0, I_{j0} - \sum_{v=1}^{s} d_{jv}\right\} \\
& + \sum_j \sum_{t=1}^{T-1} \sum_{u=t+1}^{T} \sum_{\rho=t}^{u-1} h_{j,f_{\rho+1}-1} \sum_{s \in S_t} \sum_{v \in S_u} d^n_{jv} q^{\mathrm{ext}}_{jsv} \\
& + \sum_{l,i,j,s} s_{lij} z_{lijs} + \sum_{l,j,s} \sum_{v=s}^{S} c_{lj} d^n_{jv} q_{ljsv} + \sum_{l,s} b_l (\overline{x}^{\mathrm{b}}_{ls} + \overline{x}^{\mathrm{e}}_{ls}) \\
& + \sum_{j,s} \sum_{v=s}^{S} e_j d^n_{jv} q^{\mathrm{ext}}_{jsv} + \sum_s g \cdot r_s
\end{aligned}
\tag{33}
$$

In the first two rows the inventory holding and WIP stock costs are calculated. Therefore, we differentiate three different kinds of "inflow" to the stock. The first term refers to the production quantities which are determined by multiplying the production ratios by the net demand. Here, the WIP costs are directly included, since the total shares q_{ljsv} are considered. Since items which are on stock from macroperiod t to u incur (marginal) holding costs in macroperiods $t, \ldots, u-1$, the sum over all these macroperiods has to be taken.

The second term is necessary, because for the starting inventories the full holding costs have to be charged as long as they are available. And finally, there might be items in the stock which are externally purchased. These may also incur holding costs.

The other terms on the third line of the objective function (33)—setup, production, and standby costs as well as costs for purchasing and overtime—are analogous to the original objective (1), but use the demand fractions introduced in Sect. 4.1.

Appendix B: Description of base scenarios

See Tables 7, 8 and 9.

Table 7 Data of serial base scenario (SER)

(a) Number of lines, products, macroperiods, microperiods as well as set of all last microperiods and all fixed microperiods with their respective starting times

L	4
J	6
T	4
S	12 (3 micros per macro)
Λ	{3,6,9,12}
Φ	{1,4,7,10,13}
$\overline{w}_s$	{0,–,–,80,–,–,160,–,–,240,–,–,320}

(b) Initial and maximum inventory, holding and purchasing costs as well as maximum purchase and sets of immediate and all successors

	I_{j0}	I_j^{max}	$h_{js}(s \in \Lambda)$	e_j	e_j^{max}	$\mathcal{N}_j^I$	$\mathcal{N}_j$
$j=1$	0.0	15.0	8.0	100.0	100.0	Ø	Ø
$j=2$	0.0	10.0	8.0	100.0	100.0	Ø	Ø
$j=3$	0.0	10.0	2.0	0.0	0.0	{1}	{1}
$j=4$	0.0	10.0	2.0	0.0	0.0	{2}	{2}
$j=5$	0.0	100.0	1.0	4.0	100.0	{3}	{1,3}
$j=6$	0.0	100.0	1.0	4.0	100.0	{4}	{2,4}

(c) Index sets for line synchronization constraints

$\mathcal{D}$	{(3,5,2,3), (3,6,2,4), (4,5,2,3), (4,6,2,4), (2,3,1,1), (2,4,1,2)}

Table 7 continued

(d) Overtime costs and maximum overtime

g^{max}	80
g	200

(e) Production coefficients

a_{lj}	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$	$j = 6$
$l = 1$	3.0	3.0	–	–	–	–
$l = 2$	–	–	1.0	1.0	–	–
$l = 3, 4$	–	–	–	–	4.0	3.0

(f) First microperiod and set of microperiods belonging to a macroperiod

	f_t	S_t
$t = 1$	1	{1,2,3}
$t = 2$	4	{4,5,6}
$t = 3$	7	{7,8,9}
$t = 4$	10	{10,11,12}

(g) Demand

d_{js}	$s = 3$	$s = 6$	$s = 9$	$s = 12$
$j = 1$	3.0	5.0	5.0	5.0
$j = 2$	2.0	4.0	6.0	8.0

(h) Maximum WIP

WIP_{lj}^{max}	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$	$j = 6$
$l = 1$	100.0	100.0	–	–	–	–
$l = 2$	–	–	10.0	10.0	–	–
$l = 3, 4$	–	–	–	–	0.0	0.0

(i) Setup costs and times

s_{lij}/st_{lij}		$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$	$j = 6$
$l = 1$	$i = 1$	0.0	6.0	–	–	–	–
	$i = 2$	8.0	0.0	–	–	–	–
$l = 2$	$i = 3$	–	–	0.0	2.0	–	–
	$i = 4$	–	–	12.0	0.0	–	–
$l = 3, 4$	$i = 5$	–	–	–	–	0.0	8.0
	$i = 6$	–	–	–	–	8.0	0.0

(j) Production costs

c_{lj}	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$	$j = 6$
$l = 1$	1.0	1.0	–	–	–	–

Table 7 continued

$l = 2$	–	–	1.0	1.0	–	–
$l = 3$	–	–	–	–	1.0	1.0
$l = 4$	–	–	–	–	2.0	2.0

(k) Minimum lot-sizes

m_{lj}	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$	$j = 6$
$l = 1$	1.0	1.0	–	–	–	–
$l = 2$	–	–	10.0	10.0	–	–
$l = 3, 4$	–	–	–	–	1.0	1.0

(l) Initial setup

y_{lj0}	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$	$j = 6$
$l = 1$	1	0	–	–	–	–
$l = 2$	–	–	1	0	–	–
$l = 3$	–	–	–	–	1	0
$l = 4$	–	–	–	–	0	1

(m) Bill of materials

p_{ij}	$j = 1$	$j = 2$	$j = 3$	$j = 4$
$i = 3$	6.0	–		–
$i = 4$	–	6.0	–	
$i = 5$	(6.0)	–	1.0	–
$i = 6$	–	(6.0)	–	1.0

(n) Allowed products and microperiods as well as standby costs

	$\mathcal{I}_l$	Π_l	b_l
$l = 1$	{1,2}	∅	0.0
$l = 2$	{3,4}	∅	0.0
$l = 3, 4$	{5,6}	∅	0.0

Table 8 Data of divergent base scenario (DIV)

(a) Number of lines, products, macroperiods, microperiods as well as set of all last microperiods and all fixed microperiods with their respective starting times

L	3
J	6
T	3
S	12 (4 micros per macro)
Λ	{4,8,12}
Φ	{1,5,9,13}

Table 8 continued

$\overline{w}_s$	{0,–,–,–,80,–,–,–,160,–,–,–,240}

(b) Initial and maximum inventory, holding and purchasing costs as well as maximum purchase and sets of immediate and all successors

	I_{j0}	$I_j^{\max}$	h_{js} $(s \in \Lambda)$	e_j	$e_j^{\max}$	$\mathcal{N}_j^I$	$\mathcal{N}_j$
$j = 1$	0.0	100.0	3.0	100.0	100.0	Ø	Ø
$j = 2$	0.0	100.0	3.0	100.0	100.0	Ø	Ø
$j = 3$	0.0	100.0	3.0	100.0	100.0	Ø	Ø
$j = 4$	0.0	100.0	3.0	100.0	100.0	Ø	Ø
$j = 5$	0.0	0.0	–	100.0	100.0	{1,2}	{1,2}
$j = 6$	0.0	0.0	–	100.0	100.0	{3,4}	{3,4}

(c) Index sets for line synchronization constraints

$\mathcal{D}$	{(3,5,1,1), (3,5,1,2), (3,5,2,2), (3,6,1,3), (3,6,1,4), (3,6,2,4)}

(d) Overtime costs and maximum overtime

g	200
$g^{\max}$	80

(e) Production coefficients

a_{lj}	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$	$j = 6$
$l = 1$	3.0	4.0	3.0	4.0	–	–
$l = 2$	–	8.0	–	8.0	–	–
$l = 3$	–	–	–	–	4.0	2.0

(f) First microperiod and set of microperiods belonging to a macroperiod

	f_t	S_t
$t = 1$	1	{1,2,3,4}
$t = 2$	5	{5,6,7,8}
$t = 3$	9	{9,10,11,12}

(g) Demand

d_{js}	$s = 4$	$s = 8$	$s = 12$
$j = 1$	0.0	6.0	6.0
$j = 2$	0.0	6.0	6.0
$j = 3$	2.0	6.0	6.0
$j = 4$	3.0	6.0	6.0

(h) Maximum WIP

$\text{WIP}_{lj}^{\max}$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$	$j = 6$
$l = 1$	100.0	100.0	100.0	100.0	–	–
$l = 2$	–	100.0	–	100.0	–	–
$l = 3$	–	–	–	–	0.0	0.0

Table 8 continued

(i) Setup costs and times

s_{lij}/st_{lij}		$j=1$	$j=2$	$j=3$	$j=4$	$j=5$	$j=6$
$l=1$	$i=1$	0.0	1.0	1.0	2.0	–	–
	$i=2$	2.0	0.0	3.0	1.0	–	–
	$i=3$	3.0	4.0	0.0	1.0	–	–
	$i=4$	5.0	3.0	2.0	0.0	–	–
$l=2$	$i=2$	–	0.0	–	2.0	–	–
	$i=4$	–	4.0	–	0.0	–	–
$l=3$	$i=5$	–	–	–	–	0.0	2.0
	$i=6$	–	–	–	–	6.0	0.0

(j) Production costs

c_{lj}	$j=1$	$j=2$	$j=3$	$j=4$	$j=5$	$j=6$
$l=1$	1.0	1.0	1.0	1.0	–	–
$l=2$	–	2.0	–	2.0	–	–
$l=3$	–	–	–	–	1.0	1.0

(k) Minimum lot-sizes

m_{lj}	$j=1$	$j=2$	$j=3$	$j=4$	$j=5$	$j=6$
$l=1$	1.0	1.0	1.0	1.0	–	–
$l=2$	–	1.0	–	1.0	–	–
$l=3$	–	–	–	–	1.0	1.0

(l) Initial setup

y_{lj0}	$j=1$	$j=2$	$j=3$	$j=4$	$j=5$	$j=6$
$l=1$	1	0	0	0	–	–
$l=2$	–	1	–	0	–	–
$l=3$	–	–	–	–	0	1

(m) Bill of materials

p_{ij}	$j=1$	$j=2$	$j=3$	$j=4$
$i=5$	2.0	1.0	–	–
$i=6$	–	–	2.0	1.0

(n) Allowed products and microperiods as well as standby costs

	$\mathcal{I}_l$	Π_l	b_l
$l=1$	{1,2,3,4}	Ø	0.0
$l=2$	{2,4}	Ø	0.0
$l=3$	{5,6}	Ø	0.0

Table 9 Data of base scenario with general product structure (GEN)

(a) Number of lines, products, macroperiods, microperiods as well as set of all last microperiods and all fixed microperiods with their respective starting times

L	3
J	8
T	3
S	12 (4 micros per macro)
Λ	{4,8,12}
Φ	{1,5,9,13}
$\overline{w}_s$	{0,–,–,–,80,–,–,–,160,–,–,–,240}

(b) Initial and maximum inventory, holding and purchasing costs as well as maximum purchase and sets of immediate and all successors

	I_{j0}	$I_j^{\max}$	h_{js} ($s \in \Lambda$)	e_j	$e_j^{\max}$	$\mathcal{N}_j^I$	$\mathcal{N}_j$
$j=1$	0.0	100.0	8.0	200.0	100.0	Ø	Ø
$j=2$	0.0	100.0	8.0	200.0	100.0	Ø	Ø
$j=3$	0.0	100.0	7.0	200.0	100.0	Ø	Ø
$j=4$	0.0	100.0	12.0	200.0	100.0	Ø	Ø
$j=5$	0.0	100.0	1.0	200.0	100.0	{1,2}	{1,2}
$j=6$	0.0	0.0	–	200.0	100.0	{1,3}	{1,3}
$j=7$	0.0	0.0	–	200.0	100.0	{2,4}	{2,4}
$j=8$	0.0	100.0	2.0	200.0	100.0	{3,4}	{3,4}

(c) Index sets for line synchronization constraints

$\mathcal{D}$	{(2,6,1,1), (2,6,1,3), (2,7,1,2), (2,7,1,4), (3,5,1,1), (3,5,1,2), (3,8,1,3), (3,8,1,4)}

(d) Overtime costs and maximum overtime

g	0
$g^{\max}$	0

(e) Production coefficients

a_{lj}	$j=0$	$j=1$	$j=2$	$j=3$	$j=4$	$j=5$	$j=6$	$j=7$	$j=8$
$l=1$	1.0	6.0	6.0	3.0	3.0	–	–	–	–
$l=2$	1.0	–	–	–	–	–	6.0	6.0	–
$l=3$	1.0	–	–	–	–	8.0	–	–	2.0

(f) First microperiod and set of microperiods belonging to a macroperiod

	f_t	S_t
$t=1$	1	{1,2,3,4}
$t=2$	5	{5,6,7,8}
$t=3$	9	{9,10,11,12}

Table 9 continued

(g) Demand

d_{js}	$s=4$	$s=8$	$s=12$
$j=1$	0.0	0.0	7.0
$j=2$	0.0	5.0	5.0
$j=3$	0.0	5.0	6.0
$j=4$	0.0	4.0	5.0

(h) Maximum WIP

$\text{WIP}_{lj}^{\max}$	$j=1$	$j=2$	$j=3$	$j=4$	$j=5$	$j=6$	$j=7$	$j=8$
$l=1$	200.0	200.0	200.0	200.0	–	–	–	–
$l=2$	–	–	–	–	–	200.0	200.0	–
$l=3$	–	–	–	–	200.0	0.0	0.0	200.0

(i) Setup costs and times

s_{lij}/st_{lij}		$j=0$	$j=1$	$j=2$	$j=3$	$j=4$	$j=5$	$j=6$	$j=7$	$j=8$
$l=1$	$i=0$	0.0	999.0	999.0	999.0	999.0	–	–	–	–
	$i=1$	999.0	0.0	4.0	8.0	8.0	–	–	–	–
	$i=2$	999.0	4.0	0.0	8.0	8.0	–	–	–	–
	$i=3$	999.0	6.0	6.0	0.0	4.0	–	–	–	–
	$i=4$	999.0	6.0	6.0	4.0	0.0	–	–	–	–
$l=2$	$i=0$	0.0	–	–	–	–	–	999.0	999.0	–
	$i=6$	999.0	–	–	–	–	–	0.0	4.0	–
	$i=7$	999.0	–	–	–	–	–	4.0	0.0	–
$l=3$	$i=0$	0.0	–	–	–	–	12.0	–	–	12.0
	$i=5$	0.0	–	–	–	–	0.0	–	–	8.0
	$i=8$	0.0	–	–	–	–	6.0	–	–	0.0

(j) Production costs

c_{lj}	$j=0$	$j=1$	$j=2$	$j=3$	$j=4$	$j=5$	$j=6$	$j=7$	$j=8$
$l=1$	1.0	1.0	1.0	1.0	1.0	–	–	–	–
$l=2$	1.0	–	–	–	–	–	1.0	1.0	–
$l=3$	1.0	–	–	–	–	1.0	–	–	1.0

(k) Minimum lot-sizes

m_{lj}	$j=0$	$j=1$	$j=2$	$j=3$	$j=4$	$j=5$	$j=6$	$j=7$	$j=8$
$l=1$	1.0	1.0	1.0	1.0	1.0	–	–	–	–
$l=2$	1.0	–	–	–	–	–	1.0	1.0	–
$l=3$	1.0	–	–	–	–	1.0	–	–	1.0

Table 9 continued

(l) Initial setup

y_{lj0}	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$	$j = 6$	$j = 7$	$j = 8$
$l = 1$	0	1	0	0	0	–	–	–	–
$l = 2$	0	–	–	–	–	–	1	0	–
$l = 3$	0	–	–	–	–	1	–	–	0

(m) Bill of materials

p_{ij}	$j = 1$	$j = 2$	$j = 3$	$j = 4$
$i = 5$	1.0	1.0	–	–
$i = 6$	1.0	–	0.5	–
$i = 7$	–	1.0	–	0.5
$i = 8$	–	–	1.0	1.0

(n) Allowed products and microperiods as well as standby costs

	$\mathcal{I}_l$	Π_l	b_l
$l = 1$	{0,1,2,3,4}	∅	0.0
$l = 2$	{0,6,7}	∅	0.0
$l = 3$	{0,5,8}	∅	1.0

References

Araujo S, Arenales M, Clark A (2007) Joint rolling-horizon scheduling of materials processing and lot-sizing with sequence-dependent setups. J Heuristics 13(4):337–358

Barany I, van Roy T, Wolsey L (1984) Strong formulations for multi-item capacitated lot sizing. Manag Sci 30(10):1255–1261

Buschkühl L, Sahling F, Helber S, Tempelmeier H (2010) Dynamic capacitated lot-sizing problems: a classification and review of solution approaches. OR Spectrum 32:231–261

Clark AJ, Scarf H (1960) Optimal policies for a multi-echelon inventory problem. Manag Sci 6:475–490

Dauzère-Péres S, Lasserre J-B (1994) Integration of lotsizing and scheduling decisions in a job-shop. Eur J Oper Res 75:413–426

Denizel M, Süral H (2006) On alternative mixed integer programming formulations and LP-based heuristics for lot-sizing with setup times. J Oper Res Soc 57:389–399

Denizel M, Altekin F, Süral H, Stadtler H (2008) Equivalence of the LP relaxations of two strong formulations for the capacitated lot-sizing problem with setup times. OR Spectrum 30:773–785

Drexl A, Haase K (1995) Proportional lotsizing and scheduling. Int J Prod Econ 40:73–87

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

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

Fandel G, Stammen-Hegener C (2006) Simultaneous lot sizing and scheduling for multi-product multi-level production. Int J Prod Econ 104:308–316

Ferreira D, Morabito R, Rangel S (2009) Solution approaches for the soft drink integrated production lot sizing and scheduling problem. Eur J Oper Res 196(2):697–706

Fleischmann B (1990) The discrete lot-sizing and scheduling problem. Eur J Oper Res 44:337–348

Fleischmann B, Meyr H (1997) The general lotsizing and scheduling problem. OR Spectrum 19(1):11–21

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

Haase K (1994) Lotsizing and scheduling for production planning. Springer, Berlin
Haase K (1996) Capacitated lot-sizing with sequence dependent setup costs. OR Spectrum 18:51–59
Jans R, Degraeve Z (2007) Meta-heuristics for dynamic lot sizing: A review and comparison of solution approaches. Eur J Oper Res 177:1855–1875
Kallrath J (1999) The concept of contiguity in models based on time-indexed formulations. In: Keil F, Mackens W, Voß H, Werther J (eds) Scientific computing in chemical engineering, II. Springer, Berlin, pp 330–337
Karimi B, Fatemi Ghomi S, Wilson J (2003) The capacitated lot sizing problem: a review of models and algorithms. Omega 31:365–378
Karmarkar U, Schrage L (1985) The deterministic dynamic product cycling problem. Oper Res 33(2): 326–345
Kimms A (1996) Multi-level, single-machine lot sizing and scheduling (with initial inventory). Eur J Oper Res 89(1):86–99
Krarup J, Bilde O (1977) Plant location set covering and economic lot sizing: An o(mn) algorithm for structured problems. In: Collatz L (ed) Numerische Methoden bei Optimierungsaufgaben, Bd. 3, Optimierung bei graphentheoretischen und ganzzahligen Problemen. Birkhäuser, Basel, pp 155–180
Lang JC (2009) Production and inventory management with flexible bills-of-materials and substitutions. Springer, Berlin
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 139:277–292
Meyr H (2004) Simultane Losgrößen- und Reihenfolgeplanung bei mehrstufiger kontinuierlicher Fertigung. Zeitschrift für Betriebswirtschaft 74(6):585–610
Mohammadi M, Fatemi Ghomi SMT, Karimi B, Torabi SA (2009) Development of heuristics for multi-product multi-level capacitated lotsizing problem with sequence-dependent setups. J Appl Sci 9(2):296–303
Persson J, Göthe-Lundgren M, Lundgren J, Gendron B (2004) A tabu search heuristic for scheduling the production processes at an oil refinery. Int J Prod Res 42(3):445–471
Pochet Y, Wolsey L (1994) Polyhedra for lot-sizing with Wagner–Whitin costs. Math Programm 67: 297–323
Pochet Y, Wolsey L (2006) Production planning by mixed integer programming. Springer series in operations research and financial engineering. Springer, New York
Quadt D, Kuhn H (2008) Capacitated lot-sizing with extensions: a review. 4OR: Q J Oper Res 6:61–83
Rosling K (1986) Optimal lot-sizing for dynamic assembly systems. In: Axsäter S, Schneeweiß Ch, Silver E (eds) Multi-stage production planning and inventory control. Springer, Berlin, pp 119–131
Sahling F (2010) Mehrstufige Losgrößenplanung bei Kapazitätsrestriktionen. Gabler
Salomon M, Kroon L, Kuik R, van Wassenhove L (1991) Some extensions of the discrete lotsizing and scheduling problem. Manag Sci 37:801–812
Sikora R, Chhajed D, Shaw M (1996) Integrating the lot-sizing and sequencing decisions for scheduling a capacitated flow line. Comput Ind Eng 30(4):659–679
Stadtler H (1996) Mixed integer programming model formulations for dynamic multi-item multi-level capacitated lotsizing. Eur J Oper Res 94(3):561–581
Stadtler H (2000) Improved rolling schedules for the dynamic single-level lot-sizing problem. Manag Sci 46(2):318–326
Stadtler H (2011) Multi-level single machine lot-sizing and scheduling with zero lead times. Eur J Oper Res 209(3):241–252
Stadtler H, Sahling F (2011) A lot-sizing and scheduling model for multi-stage flow lines with zero lead times. Technical report, Universität Hamburg, Institut für Logistik und Transport, Hamburg, Germany
Suerie C (2005a) Campaign planning in time-indexed model formulations. Int J Prod Res 43:49–66
Suerie C (2005b) Time continuity in discrete time models: new approaches for production planning in process industries. Springer, Heidelberg
Suerie C (2006) Modeling of period overlapping setup times. Eur J Oper Res 174(2):874–886
Suerie C, Stadtler H (2003) The capacitated lot-sizing problem with linked lot sizes. Manag Sci 49(8):1039–1054
Toledo C, França P, Morabito R, Kimms A (2009) Multi-population genetic algorithm to solve the synchronized and integrated two-level lot sizing and scheduling problem. Int J Prod Res 47(11):3097–3119

Transchel S, Minner S, Kallrath J, Löhndorf N, Eberhard U (2011) A hybrid general lot-sizing and scheduling formulation for a production process with a two-stage product structure. Int J Prod Res 49(9):2463–2480

Zhu X, Wilhelm WE (2005) Scheduling and lot sizing with sequence-dependent setup: a literature review. IIE Trans 38(11):987–1007