Lead Time Considerations for the Multi-Level Capacitated Lot-sizing Problem*

Christian Almeder[†a], Diego Klabjan[b],Renate Traxler[a], Bernardo Almada-Lobo[c]

July 18, 2014

[a]Chair for Supply Chain Management, European University Viadrina Frankfurt (Oder), Große Scharrnstr. 59, 15230 Frankfurt (Oder), Germany.

[b] Department of Industrial Engineering and Management Sciences, Northwestern University, 2145 Sheridan Road, Tech M239 Evanston, IL 60208-3119, USA.

[c]INESC-TEC, Faculdade de Engenharia da Universidade do Porto, Rua Dr. Roberto Frias s/n, Porto 4200-465, Portugal.

Abstract

The classical multi-level capacitated lot-sizing problem formulation is often not suitable to correctly capture resource requirements and precedence relations. Depending on lead time assumptions, either the model provides infeasible production plans or plans with costly needless inventory. We tackle this issue by explicitly modeling these two aspects and the synchronization of batches of products in the multi-level lot-sizing and scheduling formulation. Two models are presented; one considering batch production and the other one allowing lot-streaming. Comparisons with traditional models demonstrate the capability of the new approach in delivering more realistic results. The generated production plans are always feasible and cost savings of 30-40% compared to classical models are observed.

Keywords:_Production, Lot-sizing, Scheduling, Mixed Integer Programming, Synchronization_

[*] This research has been supported, in part, by Grant # J2815-N13 from the Austrian Science Foundation.

[†] Corresponding author:
Prof. Dr. Christian Almeder
Chair for Supply Chain Management, European University Viadrina Frankfurt (Oder)
Große Scharrnstr. 59, 15230 Frankfurt (Oder), Germany
email: Almeder@europa-uni.de, tel: +49 335 5534 2254, fax: +49 335 5534 7 2254

1 Introduction

The complexity of the product structure (bill-of-materials or BOM) of real-world cases has motivated researchers to decompose hierarchically the overall planning system into a set of more manageable subsystems. Production planning and scheduling (PPS) is one of the most prominent hierarchical systems and lies at the very heart of the performance of industrial enterprizes. At the upper level of this two-level system, the tactical driver is to determine the amount of each product that must be produced in each time-period across several resources (sizing of lots). The production scheduling (sequencing of lots) is left to the lower operational level of planning. In many situations (specially if one of the production stages represent stationary bottlenecks – see Seeanner and Meyr, 2013) this approach might be sufficient. Nevertheless, as the top level has no detailed view about system capacity in terms of scheduling, there is no guarantee that scheduling constraints (e.g., precedence, capacity) are respected, and the production plans might be infeasible (Urrutia, Aggoune, Dauzère-Pérès , 2012). In order to ensure that the plans generated at the upper-level can be scheduled afterwards, the interdependencies between both levels should be incorporated in solution approaches, through top-down and bottom-up influences.

Significant research has been conducted on the multi-level capacitated lot-sizing problem (MLCLSP) that fits the highest level of PPS. MLCLSP is considered a big time bucket model as several products can be produced in each macro period, and it is an extension of the single-level capacitated lot-sizing problem (CLSP) by incorporating the bill-of-material information (i.e. the relationship in terms of material requirements between different items to be produced) in the inventory balance constraints. Researchers have tried to consider more realistic classes of MLCLSP, including setup times, setup carryover and/or backlogging (cf. Tempelmeier and Buschkühl, 2009; Toledo et al., 2013; Wu, Shi, Geunes and Akartunali, 2011). However, even today the majority of the models and algorithms proposed for MLCLSP rely on one of the following two assumptions: either lead times are neglected, thus allowing predecessors and successors to be produced in the same period; or lead times account for at least one period for each component, forcing the throughput time (in number of periods) of the finished products to be at least equal to the number of levels of the BOM.

Buschkühl et al. (2010) report in their review that out of 16 papers dealing with metaheuristic solutions to MLCLSP, only one study (Berretta et al., 2005)considers lead time, while all others neglect it. The present work is motivated by the observation that the majority of the solutions published for well-known MLCLSP instances suffer from this assumption. Indeed, the zero lead time assumption leads to plans that are not implementable, as the lower-level scheduling problem is likely to be infeasible. This is more pronounced as the capacity tightens. On the other hand, positive lead time usually results in extensive amounts of work-in-process, which tends to increase with a larger number of levels in the

BOM.

There have been a few attempts for small-bucket approaches (in which the micro-period time structure only allows for one setup per period – see Seeanner and Meyr, 2013), but only few works include scheduling decisions within big-bucket multi-level lot-sizing (cf. Gómez Urrutia et al., 2012; Karimi-Nasab and Seyedhoseini, 2013; Stadtler, 2005).

We study the impact of not considering scheduling decisions within MLCLSP and the limitations/flaws of well known previous MLCLSP formulations. We show, by means of studying the synchronization of the batches of products within MLCLSP, that most of the solutions reported in the literature obtained by standard MLCLSP with zero lead times are infeasible, and those that consider positive lead times entail significant work-in-process. To this end, two linear mixed-integer programming formulations are proposed, one restricted to batch production and the other one allowing lot-streaming. In each time period of both models, sizing and sequencing of production lots are simultaneously addressed in such a way that precedence relationships of products are respected. The batching formulation considers production taking place in batches such that units produced within each lot can only be processed as raw materials on the subsequent BOM level once the whole batch is completed. The lot-streaming or non-batching model relaxes this requirement, allowing components to be transformed further on as soon as they are released. In the latter case, modeling is tricky and it requires a rigorous argument to establish correctness. We present results establishing correctness of the two models and comparing them with respect to the produced cost.

Both formulations are applied to benchmark instances and the differences to the classical model approaches[1] are analyzed. In contrast to MLCLSP with zero lead time the new model formulations are able to generate always feasible production plans. Comparing them with MLCLSP with positive lead time illustrates the potential cost savings of 30-40% due to a reduced total throughput time and work-in-process, respectively.

Our contributions are:

- developing two integrated models of lot-sizing and scheduling for a multi-level production system based on MLCLSP,
- proving the correctness of both formulations and establishing a relationship between them,
- analyzing the drawbacks of the classical MLCLSP model with and without lead time using standard benchmark instances,
- quantifying the potential gains by using more detailed models for such a multi-stage production planning problem.

[1] By classical model approaches we mean the multi-level lot sizing and scheduling problem without scheduling aspects as proposed by Billington et al. (1983) and described in Buschkühl et al. (2010).

In Section 2 we introduce MLCLSP and the major hurdles of its standard formulation proposed in the literature. In Section 3 two novel multi-level formulations are developed for the capacitated lot-sizing and scheduling problem, based on the assumptions for batching and lot-streaming production environments, respectively. We then address several possible extensions to these models. In Section 4 we then apply the new model formulation to standard benchmark instances and compare the results with those of the classical MLCLSP.

Related Literature

Lot-sizing problems have been studied by researchers in different variations during the last decades. One of the initial works describing the trade-off between the setup and holding costs for a dynamic demand scenario is provided by Wagner and Within (1958). Introducing capacity limitations and considering several products simultaneously led to the capacitated lot-sizing problem (CLSP), which Bitran and Yanasse (1982) have shown to be NP-hard. In the case of positive setup times, Maes et al. (1991) proved that already the feasibility problem is NP-complete. MLCLSP dealt in this paper was first introduced by Billington et al. (1983) and it extended the lot-sizing models towards material requirements planning. Before this work, lot-sizing problems have mainly been applied at the final product level.

A recent review about different model formulations and solution methods for MLCLSP can be found in Buschkühl et al. (2010). This review points out that it is necessary to consider manufacturing lead times, but many researchers neglect them. For example, 15 out 16 papers on metaheuristic solution methods for MLCLSP mentioned in that review neglect lead time (e.g. Berretta and Rodrigues, 2004; Pitakaso et al., 2006; Xie and Dong, 2002). The remaining work Berretta et al. (2005) assumes that some items have no lead times and others have positive lead times of one period or more. More recently, also Wu, Shi, Geunes and Akartunali (2011) and Wu, Shi, Geunes and Akartunali (2012) present MIP models for solving capacitated multi-level lot-sizing problems with setup times and backlogging, considering null lead times. Clark et al. (2011) stress the need to incorporate more specificities of production environments in lot-sizing models, including the synchronization of resources.

The present paper deals with the manufacturing lead time consideration for MLCLSP. This involves also scheduling decisions. The classical MLCLSP problem is considered to be a big-bucket model, as the planning horizon is divided into a small number of long time periods allowing for multiple products/setups per time bucket. On the other hand, the class of small-bucket models try to capture both lot-sizing and scheduling decisions, as at most one setup might be performed in each period and, consequently, a complete production sequence is generated by default, Drexl and Kimms (1997). Here, the planning horizon is divided into many short periods. Nevertheless, Wolsey (2002) provides a comprehensive study and classification scheme for different small- and big-bucket models, and shows that the LP relaxation of small-bucket models usually delivers very weak lower bounds. Only with customized reformulations and valid inequalities added to the problem an improvement of the lower bound is possible. In contrast, most big-bucket models (MLCLSP is classified as such) provide much better lower bounds (see Almeder and Almada-Lobo, 2011).

A few researchers have tried to incorporate the multi-level structure and the exact timing issues into small-bucket models. Stadtler (2011) presents a proportional lot-sizing and scheduling problem formulation for a multi-level problem. In order to allow synchronization of the different levels within a period, the problem is restricted to a single capacitated resource, i.e., it is not possible to produce different items simultaneously on different machines. Meyr (2004) and Seeanner and Meyr (2013) use a general lot-sizing and scheduling problem formulation (GLSP) to model a multi-level system with possible simultaneous production of predecessor and successor items. The GLSP is based on time buckets of variable length (usually determined by the size of a lot). In order to allow synchronization, all machines must use the same period structure, which might lead to unproductive idle times. Furthermore, it is not possible to enforce the finishing of a production lot before the production of a successor starts.

For the class of big-bucket lot-sizing models, there are extensions to incorporate scheduling decisions. Haase (1996), Gupta and Magnusson (2005), and Almada-Lobo et al. (2008) developed extensions for the single-level capacitated lot-sizing problem to deal with sequence dependent setups. In what concerns big-bucket models for the multi-level setting, we are only aware of one paper (Fandel and Stammen-Hegene, 2006) that tackles this problem. The authors model a similar problem as our here, but in contrast to our approach, the authors develop a non-linear formulation. The complexity of that formulation prohibits to solve even small instances. To the best of our knowledge, efficient solution methodologies for their model are not known as of this date. The review by Zhu and Wilhelm (2006) analyzes the research on the intersection of lot-sizing and scheduling from the scheduling point of view.

Another subject related to this work is the problem of synchronizing the use of common resources. Tempelmeier and Buschkühl (2008) describe a problem where a common setup operator has to perform the setup operations on different machines. Hence, it is necessary to ensure that there are no two setup operations at the same time. In Almeder and Almada-Lobo (2011) a model describing the synchronization of a secondary resource used in a parallel machine environment is developed.

2 Motivation

In this paper, we study extensions to the following classical MLCLSP model as formulated by Billington et al. (1983).

2.1 Classical MLCLSP formulation

We assume a production planning problem with N items (including finished goods, subassemblies, and raw materials), T time periods and M machines and the underlying BOM. Each item has a manufacturing lead time and deterministic demand in each time period. The problem is to find production quantities in each time period that obey to BOM requirements, demand requirements, limited capacity resources, and simultaneously minimizing the production and holding costs. It reads

$$\min \sum_{i=1}^{N} \sum_{t=1}^{T} (c_i \cdot Y_{it} + h_i \cdot I_{it}) \tag{1}$$

subject to

$$I_{it} = I_{i(t-1)} + X_{i(t-l_i)} - \sum_{j \in \Gamma(i)} a_{ij} \cdot X_{jt} - E_{it} \qquad i,t \tag{2}$$

$$\sum_{i=1}^{N} (p_{mi} \cdot X_{it} + s_{mi} \cdot Y_{it}) \leq L_{mt} \qquad m,t \tag{3}$$

$$X_{it} - G \cdot Y_{it} \leq 0 \qquad i,t \tag{4}$$

$$I_{it} \geq 0, X_{it} \geq 0, Y_{it} \in \{0,1\} \qquad i,t \tag{5}$$

and with the decision variables:

I_{it} inventory level of item i at the end of period t

X_{it} production amount of item i in period t

Y_{it} $\begin{cases} 1 & \text{if item } i \text{ is produced in period } t \\ 0 & \text{otherwise.} \end{cases}$

The parameters are as follows:

a_{ij} quantity of item i required to produce one unit of item j (*gozinto-factor*)

p_{mi} time for producing one unit of item i on machine m

c_i setup cost of item i

E_{it} (external) demand of item i in period t

G arbitrary large number (e.g. total demand or maximum capacity)

I_{i0} initial inventory level of item i

h_i holding cost of item i

L_{mt} available capacity (time) of machine m in period t

l_i lead time of item i (nonnegative integer corresponding to the number of periods)

s_{mi} time for setting up machine m for the production of item i

$\Gamma(i)$ set of immediate successors of item i based on BOM.

The index set (i,j,t,m) is defined by $i,j \in \{1,2,\ldots,N\}$, $t \in \{1,2,\ldots,T\}$ and $m \in \{1,2,\ldots,M\}$. Objective function (1) captures the fixed setup cost and the underlying holding cost. Constraints (2) represent the standard lot-sizing inventory balance capturing BOM and lead times. Limited machine capacity is reflected by (3) and (4) captures the definition of setup variables.

Many researchers dealing with this model assume that manufacturing lead times are negligible, i.e. predecessors and successors might be produced in the same period ($l_i = 0$ for every i). Since MLCLSP is a big-bucket model and periods are assumed to cover long time slots with several different production batches, the exact scheduling process is usually postponed to the next planning level (the bottom level of the PPS hierarchical system). Assuming that the capacity utilization is not too high and/or that there is always the possibility of using overtime to compensate a potential lack of capacity, it might be reasonable to neglect the lead time. Nevertheless, in case of capacity tight scenarios and of a limited usage of overtime (which happens for example in a production system running 24 hours a day, 7 days a week) this zero lead time assumption may lead to plans that are not implementable in practice.

On the other hand, positive lead times of at least one period ($l_i = 1$ for every i) do deliver feasible solutions that can be implemented. But positive lead times contradict somehow the big-bucket assumption, because in practice they would lead to extensive amounts of work-in-process. For example, consider a time period of one day and an average production batch of 3 hours. For a BOM with 10 levels it would take ten days to finish a final item according to MLCLSP, but less than two days in practice. Note, that in the case of positive lead times the objective function (1) does not capture the additional work-in-process. An extra constant cost term depending on the total demand of all periods has to be added.

2.2 Drawbacks of MLCLSP

Let us assume a simple example with four items (1, 2, 3, and 4), three machines (A, B, and C), and two periods. Item 1 is produced on machine A, item 2 on machine B and items 3 and 4 on machine C. Items 1 and 2 are final items, where one unit of item 3 is necessary to produce one unit of item 1, and one unit of item 4 is necessary to produce one unit of item 2 and one unit of item 3, respectively. Table 1 contains all parameters and Figure 1 depicts the product structure. It is assumed that resource requirements are measured in time units, each machine has a capacity of 1 (meaning it is available throughout the whole period) and setup times are 0.05 for all products. Furthermore, we assume that machine A is initially setup for item 1, machine B for item 2, and machine C for item 4.

The optimal solution of the MLCLSP model without lead time can be easily obtained (see Table 2). The total cost is 22 consisting of the setup cost of 20 (four setups) and the holding cost of 2 (two units of item 4 are carried to period 2). It is obvious that this solution is not feasible in practice, because before item 1 can be produced in the first period, items 3 and 4 have to be scheduled. Figure 2(a) shows the production plan generated by scheduling

Table 1: Data for the example

item	machine	E_{i1}	E_{i2}	h_i	c_i	p_{mi}	s_{mi}
1	A	3	0	3	5	0.1	0.05
2	B	0	2	2	5	0.4	0.05
3	C	0	0	2	5	0.1	0.05
4	C	0	0	1	5	0.1	0.05

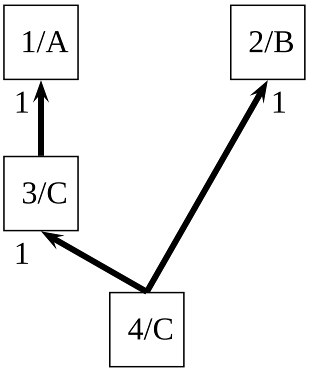
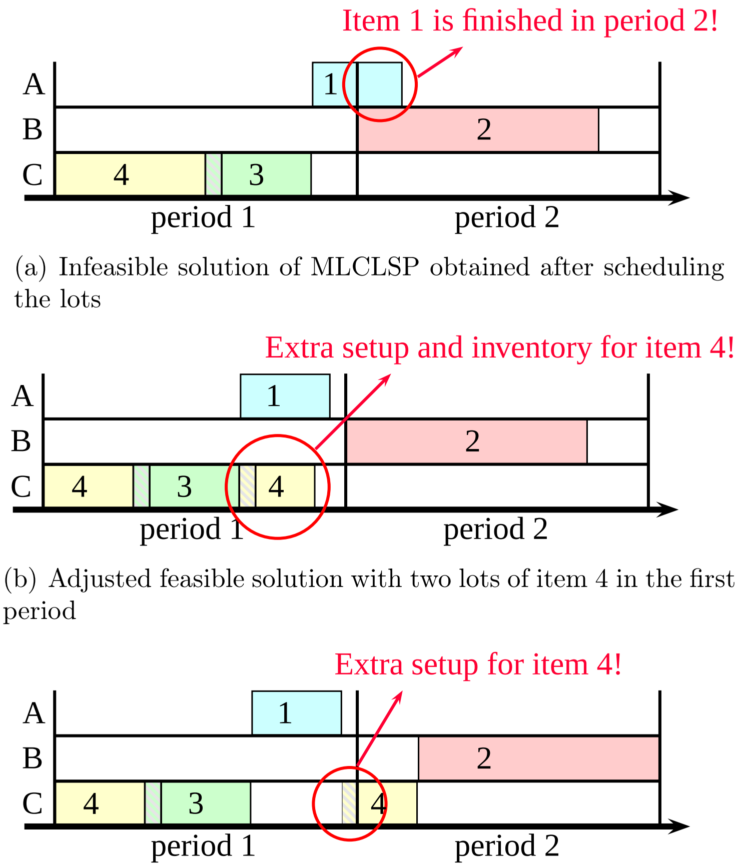


Figure 1: BOM for the example

the solution obtained by the MLCLSP formulation. Note, that MLCLSP does not consider the initial setup state of the machines. Therefore the solution depicted in Figure 2(a) saves the cost and time of three setups (if machines A, B, and C are initially set up for items 1, 2, and 4) and has a total cost of only 7 (one setup from item 4 to item 3 and two units of item 4 carried to period 2). But obviously this production plan violates the demand requirement, as the lot of item 1 is finished in period 2 and not in period 1. The second production plan (Figure 2(b)) is generated from the first one, by splitting the lot of item 4 such that item 1 can be finished on time. But an additional setup for item 4 increase the cost to 13. Furthermore, this version would violate the implicit assumption of the traditional lot-sizing model that each item is produced at most once in a period. The third plan (Figure 2(c)) represents the optimal feasible solution of a simultaneous lot-sizing and scheduling model. The second batch of item 4 is moved to the second period (still the setup operation has to be done at the end of period 1), ending up with a total cost of 10 (two setups on machine C and no holding cost). But this solution is in essence different from that of MLCLSP. Clearly, if we consider one period lead time for MLCLSP, the problem is infeasible.

Table 2: MLCLSP solution with no lead time for the example

item	x_{i1}	x_{i2}	I_{i1}	I_{i2}
1	3	0	0	0
2	0	2	0	0
3	3	0	0	0
4	5	0	2	0



Item 1 is finished in period 2!

(a) Infeasible solution of MLCLSP obtained after scheduling the lots

Extra setup and inventory for item 4!

(b) Adjusted feasible solution with two lots of item 4 in the first period

Extra setup for item 4!

(c) Optimal solution of the problem

Figure 2: Gantt charts after scheduling the lots obtained by MLCLSP and possible adjustments for obtaining a feasible production plan. The different bars indicate the actual production times for the items. Shaded bars represent setup operations.

3 Models

In order to address the above mentioned pitfalls of MLCLSP in a systematic way and to come up with an alternative, we suggest to embed scheduling to MLCLSP, i.e., to consider the starting times of operations and the precedence relations in production between products in an integrated manner. We present two model formulations. The first one is based on the batching assumption, i.e., the production of a successor item can only start if the complete batch of the predecessor item is finished. For the second formulation, the lot-streaming case, we relax this constraint and allow simultaneous production of predecessor and successor items.

3.1 Batching Production

This section develops a multi-level capacitated lot-sizing formulation that schedules the production lots and synchronizes the batches of predecessors and successors, based on (1)-(5). As for the MLCLSP model formulated before, we assume that each item is assigned to a single machine, which is the case for most problems published and test instances available (see Tempelmeier and Derstroff, 1996). The following additional parameters are introduced:

$\phi(m)$ set of items that can be assigned to machine m

p_i time for producing one unit of item i (we can omit index m because each item is assigned to a specific machine)

s_{ij} setup time for a changeover from product i to product j, $s_{ii} = 0$ for each i

c_{ij} setup cost for a changeover from product i to product j, $c_{ii} = 0$, $i, j \in \phi(m)$

We need the following decision variables:

μ^s_{it} start time of the production of item i in period t

$\hat{X}_{ijt}$ production amount of item $j \in \Gamma(i)$ starting before production of item i is finished in period t

$$\alpha_{itm} \quad \begin{cases} 1 & \text{if machine } m \text{ is set up for item } i \text{ at the beginning of period } t \\ 0 & \text{otherwise} \end{cases}$$

$$T_{ijtm} \quad \begin{cases} 1 & \text{if there is a changeover from item } i \text{ to } j \text{ on machine } m \text{ in period } t \\ 0 & \text{otherwise} \end{cases}$$

$$W_{ijt} \quad \begin{cases} 1 & \text{if production of item } j \text{ starts after the whole batch of item } i \\ & \text{is completed in period } t \text{ and } j \in \Gamma(i) \\ 0 & \text{otherwise.} \end{cases}$$

By scaling, we assume without loss of generality that the duration of a period is a single time unit (i.e., $L_{mt} = 1$ for every m and t). Hence, all time-related parameters and decision variables are scaled to this time unit. We first consider the case where the production takes place in batches. In other words, at the start of the production of item i the whole amount of necessary predecessors (raw materials) must be available and the finished items are available for other production processes only after the whole batch is finished. Hence, the stock level of item i is a discontinuous step function (see Figure 3) where a decrease in stock is due to the consumption of item i by a different item and an increase in inventory results from a finished batch.

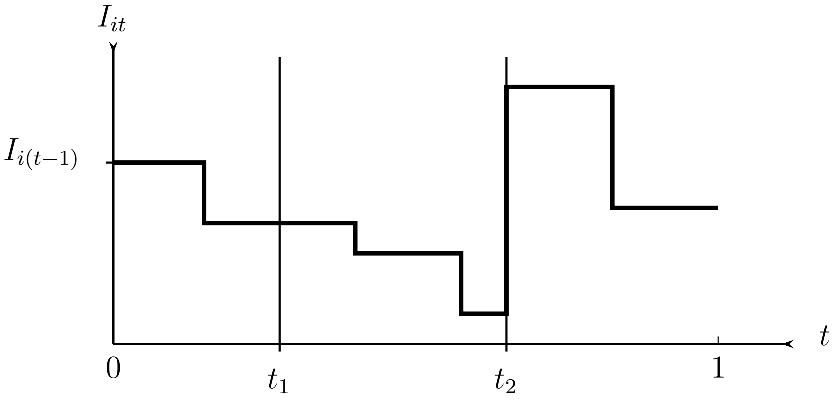


Figure 3: Change of the available stock of item i: t_1 denotes the start and t_2 the finish time of the production of item i.

Note, that $T_{ijtm} = 0$ for every $i \notin \phi(m)$ or $j \notin \phi(m)$, and for $i = j$. We also have $\alpha_{itm} = 0$ for every $i \notin \phi(m)$. Let $\nu(\cdot)$ denote the optimal value of the underlying optimization problem. The corresponding lot-sizing and scheduling problem for the batching production (F_B) reads:

$$\nu(F_B) = \min \sum_{i=1}^{N} \sum_{t=1}^{T} h_{it} \cdot I_{it} + \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{t=1}^{T} \sum_{m=1}^{M} c_{ij} \cdot T_{ijtm} \tag{6}$$

subject to

$$I_{it} - I_{i(t-1)} - X_{it} + \sum_{j \in \Gamma(i)} a_{ij} \cdot X_{jt} + E_{it} = 0 \qquad i, t \tag{7}$$

$$p_i \cdot X_{it} \leq \sum_{j \in \phi(m)} T_{jitm} + \alpha_{itm} \qquad i \in \phi(m), t, m \tag{8}$$

$$\sum_{i \in \phi(m)} \alpha_{itm} = 1 \qquad t, m \tag{9}$$

$$\sum_{j \in \phi(m)} T_{jitm} + \alpha_{itm} = \sum_{j \in \phi(m)} T_{ijtm} + \alpha_{i(t+1)m} \qquad i \in \phi(m), t, m \tag{10}$$

$$\sum_{j \in \phi(m)} T_{jitm} \leq 1 \qquad i \in \phi(m), t, m \tag{11}$$

$$\mu^s_{it} + p_i \cdot X_{it} + s_{ij} \cdot T_{ijtm} + T_{ijtm} - 1 - \alpha_{j(t+1)m} \leq \mu^s_{jt} \qquad i \in \phi(m), j \in \phi(m), t, m \tag{12}$$

$$\mu^s_{it} + p_i \cdot X_{it} + s_i j \cdot T_{ijtm} + \alpha_{j(t+1)m} - 1 - \alpha_{jtm} \leq \mu^s_{jt} \qquad i \in \phi(m), j \in \phi(m), t, m \tag{13}$$

$$\mu^s_{it} + p_i \cdot X_{it} + \sum_{j \in \phi(m)} s_{ij} \cdot T_{ijtm} \leq 1 \qquad i \in \phi(m), t \tag{14}$$

$$(\mu^s_{it} + p_i \cdot X_{it}) - \mu^s_{jt} \leq 1 - W_{ijt} \qquad i, j \in \Gamma(i), t \tag{15}$$

$$\hat{X}_{ijt} \geq X_{jt} - \frac{1}{p_j} W_{ijt} \qquad i, j \in \Gamma(i), t \tag{16}$$

$$I_{i(t-1)} \geq \sum_{j \in \Gamma(i)} a_{ij} \hat{X}_{ijt} \qquad i, t \tag{17}$$

$$I_{it}, X_{it}, \hat{X}_{ijt}, \mu^s_{it} \geq 0; \alpha_{itm}, T_{ijtm}, W_{ijt} \in \{0, 1\} \quad i, j, t, m. \tag{18}$$

Objective function (6) aims at minimizing the sum of sequence-dependent setup and holding costs. Constraints (7) represent the inventory balance for both components and end-items. Constraints (8) guarantee that a product is produced only if the machine is set up for it. Note that we assume that the length of one period is one time unit and capacities are measured in time units. Therefore, in contrast to constraints (4), we can omit parameter G in this case. Based on (9) each machine has to be set up for exactly one product at the beginning of each period. Constraints (10) carry the setup configuration state of the machines into the next period. They represent a flow conservation for the setup state. So if a machine is set up for a certain item i in period t, there must be either a changeover from item i to another item ($\sum_{j\in\phi(m)} T_{ijtm}$) or the setup state is carried over to the next period ($\alpha_{i(t+1)m}$). Constraints (11) avoid more than one setup for each item. The production sequence on a certain machine within a period represents an open tour including some of the items. As in vehicle routing problems we have to avoid disconnected subtours, i.e., setup sequences which cycle between the same products within a period (e.g., $i_1 \rightarrow i_2 \rightarrow \cdots \rightarrow i_1$) but where the initial setup state of the machine is not part of this sequence. Constraints (12) cut off such disconnected subtours and subtours that appear in the middle of the sequence (see e.g., Almada-Lobo et al., 2007 or Almada-Lobo et al., 2008 for further details). Furthermore, these constraints link the starting and finishing times of two batches produced one immediately after the other. We do not impose the starting times of the first batch in any period to be zero, as there might be some idle time before production starts (as well as afterwards). Requirements (12) do not define the starting time of the last product to be produced on each machine, as they are inactive for such cases. We assume that for a cycle with more than one item involving item j ($\alpha_{jtm} = \alpha_{j(t+1)m} = 1$), there occurs only one batch of j in the first position of the production sequence. In case of a path, requirements (13) determine the starting time of the last product of the sequence. Clearly, (13) are redundant for a cycle. Inequalities (12)–(14) ensure that the machines are used no longer than their available capacity. Constraints (15) ensure the coherency between variables W_{ijt} and the start and finish times of the batches of predecessors and successors. Note that $\mu^s_{it} + p_i \cdot X_{it}$ computes the production completion time of product i in period t. In case item j starts the production before the batch of its predecessor i is completed in the same period (or in case there is not such batch), i.e. $W_{ijt} = 0$, constraints (16) and (17) force the quantity of item i required to produce j to be supplied solely from the initial stock in that period. If $W_{ijt} = 1$, then (16) and (17) are inactive, and (7) allow the gross requirements of item i to produce j to be matched from stock and from production in the same period.

Note that the way how depended batches are synchronized is motivated by a model formulation given in Almeder and Almada-Lobo (2011) where the usage of a secondary resource is synchronized across parallel machines.

Example continued. For a better understanding of the model let us return to the example given in Section 2.2. Figure 4 depicts the optimal solution obtained by the model above with total cost of 10. The positive values of the decision variables are indicated at the different positions. All decision variables not mentioned have a value of 0.

The variables α_{itm} and T_{ijtm} determine the changes of the setup state of the machines and the items which may be produced. The values of α_{i1m} correspond to the given initial setup state of the machines. Since there is no setup on machines A and B, in period 1 variables T_{ijtm} equal to 0 for $m = A, B$. On machine C we have in the first period a changeover from

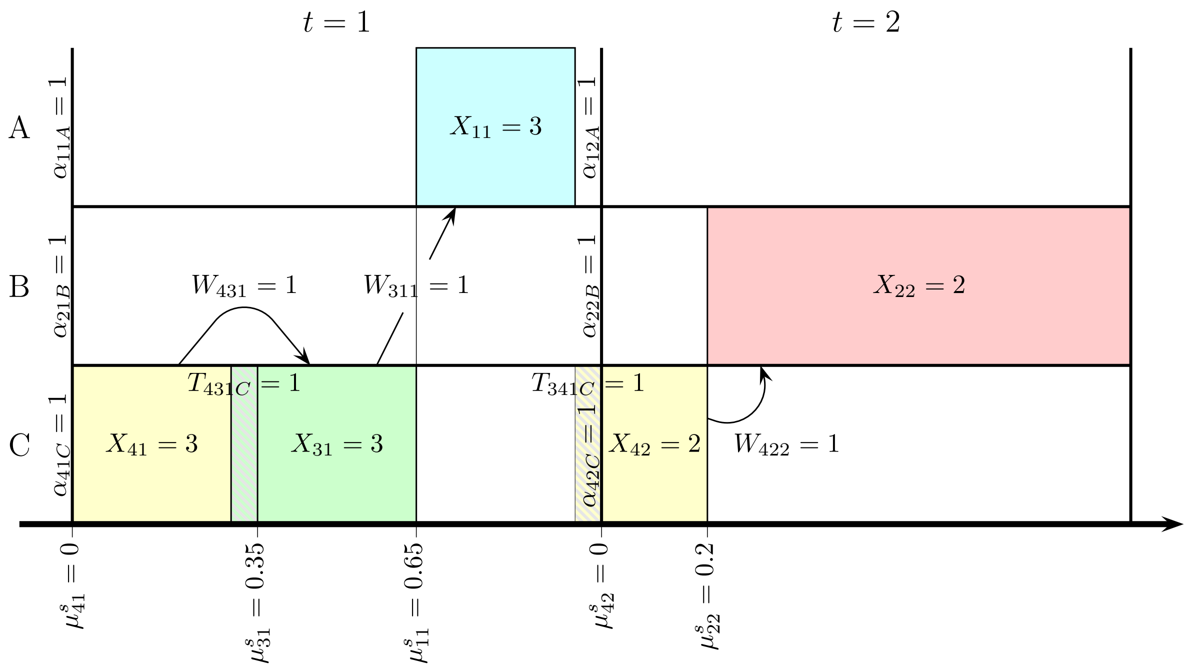


Figure 4: Illustration of decision variables for the batching case

item 4 to item 3 and back to item 4. Hence, for items 3 and 4 constraints (10) are as follows:

$$\underset{T_{431C}}{1} + \underset{\alpha_{31C}}{0} = \underset{T_{341C}}{1} + \underset{\alpha_{32C}}{0}$$

$$\underset{T_{341C}}{1} + \underset{\alpha_{41C}}{1} = \underset{T_{431C}}{1} + \underset{\alpha_{42C}}{1}$$

The constraints (12) and (13) ensure a correct production sequence within a period and enforce the decision variables μ^s_{it} to take the correct starting times of the lots. Let us look in more detail to constraints (12) for items 3 and 4 in period 1:

$$\underset{\mu^s_{31}}{0.35} + \underset{p_3 \cdot X_{31}}{0.3} + \underset{s_{34} \cdot T_{341C}}{0.05} + \underset{T_{341C}}{1} - 1 - \underset{\alpha_{42C}}{1} = -0.3 \leq \underset{\mu^s_{41}}{0}$$

$$\underset{\mu^s_{41}}{0} + \underset{p_4 \cdot X_{41}}{0.3} + \underset{s_{43} \cdot T_{431C}}{0.05} + \underset{T_{431C}}{1} - 1 - \underset{\alpha_{32C}}{0} = 0.35 \leq \underset{\mu^s_{31}}{0.35}$$

The last term of the left-hand-side (α_{jtm}) is necessary to allow a changeover back to item 4 at the end of period 1. Otherwise, the two constraints above would conflict and no feasible values for μ^s_{31} and μ^s_{41} would be possible. In case there is no changeover back to the first item produced in that period, constraints (13) is turned active enforcing the correct starting value of the last lot in a period.

Since in period 1 the lot of item 3 is produced after item 4 and item 1 after item 3, constraints (15) do not enforce W_{ijt} to be 0 for those item combinations:

$$\underset{\mu^s_{41}+p_4\cdot X_{41}}{0.3} - \underset{\mu_{31}}{0.35} = -0.05 \leq 1 - W_{431}$$
$$\underset{\mu^s_{31}+p_3\cdot X_{31}}{0.65} - \underset{\mu_{11}}{0.65} = 0 \leq 1 - W_{311}$$

Hence, W_{431} and W_{311} can take the values 1, such that $\hat{X}_{431} = \hat{X}_{311} = 0$ according to (16) and consequently, due to constraints (17) there is no inventory of item 4 or 3 necessary before the start of the period. □

Next we show that formulation F_B really models the problem correctly. Let Q be the set of solutions satisfying constraints (7)–(15) and (18). Set Q admits solutions that have nonnegative inventory stock for every product at the end of every period (imposed by (7)), but potentially negative in-period stock. We show that constraints (16) and (17) avoid this case. To this end, consider subsets $S(i)$ of items in a given time period t defined as

$$S(i) = \{j \in \Gamma(i) : \mu^s_{jt} < \mu^s_{it} + p_i \cdot X_{it}\}.$$

The following lemma states that there are no negative in-period stocks (note that due to the batching assumption the condition $\sum_{j\in S(i)} a_{ij} \cdot X_{jt} > I_{i(t-1)}$ is equivalent to the statement of having negative in-period stock).

Lemma 1. *If $x^\star \in Q$ and $x^\star$ fulfills (16) and (17), then $\sum_{j\in S(i)} a_{ij} \cdot X_{jt} \leq I_{i(t-1)}$.*

Proof. Let $x^\star \in Q$. We first observe that if $j \in S(i)$, then $\mu^s_{jt} < \mu^s_{it} + p_i \cdot X_{it}$, which in turn from (15) implies $W_{ijt} = 0$. Thus if $j \in S(i)$, from (16) we obtain $\hat{X}_{ijt} \geq X_{jt}$.

On the other hand, if $j \in \Gamma(i)\backslash S(i)$, then clearly $\hat{X}_{ijt} \geq 0$. Combining these two facts we derive

$$I_{i(t-1)} \geq \sum_{j\in\Gamma(i)} a_{ij} \cdot \hat{X}_{ijt} \geq \sum_{j\in S(i)} a_{ij} \cdot \hat{X}_{ijt} + \sum_{j\in\Gamma(i)\backslash S(i)} a_{ij} \cdot \hat{X}_{ijt} \geq \sum_{j\in S(i)} a_{ij} \cdot \hat{X}_{ijt} \geq \sum_{j\in S(i)} a_{ij} \cdot X_{jt}.$$

This establishes the lemma. □

In addition to showing that solutions have no negative in-period stock, we need to assert that no solutions to the problem are left out.

Lemma 2. *Any $x^\star \in Q$ satisfying $\sum_{j\in S(i)} a_{ij} \cdot X_{jt} \leq I_{i(t-1)}$ for every i and t (x^* has nonnegative in-period inventory) fulfills constraints (16) and (17).*

Proof. We show this by contradiction. We first note that in (16) and (17) we have the freedom of selecting $\hat{X}_{ijt}$. As a result we can assume that we have an underlying $\hat{X}_{ijt}$ satisfying (16) but not (17) for at least one item i' in a period t' since otherwise we could always increase $\hat{X}_{ijt}$ to satisfy (16), but possibly violate (17) even more. Thus we have

$$\begin{aligned} \hat{X}_{i'jt'} &\geq X_{jt'} - \frac{1}{p_j} W_{i'jt'} && j \in \Gamma(i') \\ I_{i'(t'-1)} &< \sum_{j \in \Gamma(i')} a_{i'j} \hat{X}_{i'jt'}. \end{aligned}$$

Among all such $\hat{X}_{i'jt'}$, we pick the smallest one, which we denote by $\hat{X}^{\min}_{i'jt'}$. By choice we clearly have

$$I_{i'(t'-1)} < \sum_{j \in \Gamma(i')} a_{i'j} \hat{X}^{\min}_{i'jt'}. \tag{19}$$

According to the definition of $S(i)$, the minimal values satisfying (16) are defined by

$$\hat{X}^{\min}_{i'jt'} = \begin{cases} X_{jt'} & j \in S(i') \\ 0 & j \in \Gamma(i') \backslash S(i'). \end{cases}$$

Inequality (19) yields

$$I_{i'(t'-1)} < \sum_{j \in \Gamma(i')} a_{i'j} \hat{X}^{\min}_{i'jt'} = \sum_{j \in S(i')} a_{i'j} \hat{X}^{\min}_{i'jt'} + \sum_{j \in \Gamma(i') \backslash S(i')} a_{i'j} \hat{X}^{\min}_{i'jt'} = \sum_{j \in S(i')} a_{i'j} X_{jt'}.$$

This constraint indicates that there is negative in-period inventory and it contradicts our assumption that $\sum a_{i'j} \cdot X_{jt} \leq I_{i'(t'-1)}$. □

3.2 Lot-streaming Production (Non-batching)

The previous model assumed that components can only be used for the production of successors if the production batch is completely finished. If this assumption is relaxed, we allow to use some already finished parts of the current production batch for the production of successors. No transportation times are considered between different production stages and inventory levels increase and decrease continuously due to production and consumption of an item. It has to be ensured that during the whole period the inventory level is nonnegative. Instead of constraints (16) and (17) it is necessary to replace them with

$$I_{i(t-1)} + \min\left(X_{it}, \frac{1}{p_i}\left(\tau - \mu^s_{it}\right)^+\right) \geq \sum_{j \in \Gamma(i)} a_{ij} \min\left(X_{jt}, \frac{1}{p_j}\left(\tau - \mu^s_{jt}\right)^+\right) \quad i, t, \tau \in [0, 1], \tag{20}$$

where $x^+ := \max(x, 0)$.

This set of constraints ensures that, for every time point τ in period t, the initial inventory plus the production of item i until τ is not less than the consumption of item i due to the production of successors. Note that the external demand E_{it} only occurs at the end of the period. Unfortunately there is an infinite number of such constraints. The next result asserts that only a finite number of τ values suffices.

Theorem 1. *It is sufficient and necessary to fulfill constraints (7) and (20) at the start of the production batch of item i ($\tau = \mu^s_{it}$) and at the end of the production batch of each successor $k \in \Gamma(i)$ ($\tau = \mu^s_{kt} + p_k X_{kt}$) in order to maintain a nonnegative inventory level throughout the whole period t.*

Proof. We fix item i. Let us first consider the case when there is no production of item i in period t ($X_{it} = 0$). If there is no production, the in-period inventory level of item i is non-increasing throughout the period. Hence it reaches the minimum at the end of the period. Since constraints (7) ensure that there is a nonnegative stock level at the end of each period, the in-period inventory level can never be negative. To show the other way around, because of (7), the following derivation shows that (20) is fulfilled for all τ:

$$I_{i(t-1)} + \min\left(X_{it}, \frac{1}{p_i}\left(\tau - \mu^s_{it}\right)^+\right) = I_{i(t-1)} = I_{it} - X_{it} + E_{it} + \sum_{j\in\Gamma(i)} a_{ij}X_{jt} \geq$$

$$\geq \sum_{j\in\Gamma(i)} a_{ij}X_{jt} \geq \sum_{j\in\Gamma(i)} a_{ij}\min\left(X_{jt}, \frac{1}{p_j}\left(\tau - \mu^s_{jt}\right)^+\right).$$

Now let us consider the case of a production of item i in period t ($X_{it} > 0$). We divide the period into three parts. Let t_1 define the start of the production of item i ($t_1 = \mu^s_{it}$), and t_2 the end of the production of item i ($t_2 = \mu^s_{it} + p_i X_{it}$). Furthermore, note that $\frac{1}{p_i}$ is the production rate of item i, $\frac{a_{ij}}{p_j}$ is the consumption rate of item i due to the production of successor j, and the in-period inventory level $I_{it}(\tau)$ is a piecewise linear function of τ.

Let first $\tau \in [0, t_1]$. During this period only consumption of item i might occur ($I_{it}(\tau)$ is non-increasing), so it is sufficient and necessary that at time t_1 the inventory level is nonnegative, which is equivalent to constraints (20) at $\tau = t_1 = \mu^s_{it}$.

Consider now $\tau \in [t_2, 1]$. After t_2 again only consumption might occur and the in-period inventory level is non-increasing ($I_{it}(\tau)$ is again non-increasing). For the time span from t_2 to the end of the period it is sufficient and necessary to have nonnegative inventory at the end of the period. This is assured by constraints (7).

Finally, let $\tau \in [t_1, t_2]$. During the production period of item i between t_1 and t_2 the in-period inventory level may increase or decrease depending on the difference between the constant production rate $\frac{1}{p_i}$ and the consumption rate of all currently processed successors (see Figure 5). This consumption rate might vary depending on the number of successors produced at a specific time point. If we consider a specific successor k of item i, during the production of item k, the consumption rate of item i is increased by $\frac{a_{ik}}{p_k}$. Since the consumption rate due to item k stays constant over the whole production period $[t_1, t_2]$, it is sufficient and necessary that at the end of this production period the in-period inventory level is nonnegative, i.e. $I_{it}(\tau) \geq 0$ for $\tau = \mu^s_{kt} + p_k X_{kt}$. We need to guarantee this for all k and thus $I_{it}(\tau) \geq 0$ for $\tau \in \{\mu^s_{kt} + p_k X_{kt} | k \in \Gamma(i) \text{ and } t_1 \leq \mu^s_{kt} + p_k X_{kt} \leq t_2\}$. Using the same argument as in the case of no production it can be shown that (20) for $\tau < t_1$ and $\tau > t_2$ are valid and not cutting off solutions with nonnegative inventory. Hence, (20) must hold for $\tau \in \{\mu^s_{kt} + p_k X_{kt} | k \in \Gamma(i)\}$. □

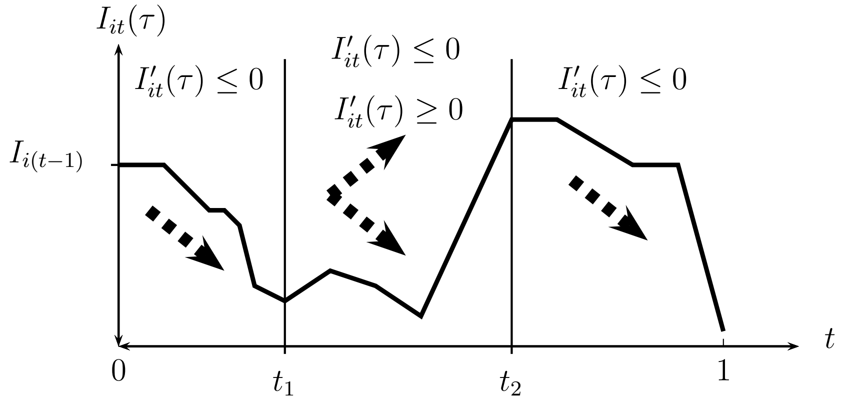


Figure 5: Example of the in-period inventory

From Theorem 1 we conclude that replacing (16) and (17) with a set of constraints avoiding negative stock levels at the beginning and end of the production of each item i for each successor k by

$$I_{i(t-1)} \geq \sum_{j \in \Gamma(i)} a_{ij} \min\left(X_{jt}, \frac{1}{p_j}\left(\mu^s_{it} - \mu^s_{jt}\right)^+\right) \qquad i, t \qquad (21)$$

$$I_{i(t-1)} + \min\left(X_{it}, \frac{1}{p_i}\left(\mu^s_{kt} + p_k X_{kt} - \mu^s_{it}\right)^+\right) \geq$$
$$\sum_{j \in \Gamma(i)} a_{ij} \min\left(X_{jt}, \frac{1}{p_j}\left(\mu^s_{kt} + p_k X_{kt} - \mu^s_{jt}\right)^+\right) \qquad i, k \in \Gamma(i), t \qquad (22)$$

leads to a model that allows immediate use of produced items but maintains a nonnegative in-period inventory.

Hence, the model for the lot-streaming case F_L reads:

$$\nu(F_L) = \min \sum_{i=1}^{N} \sum_{t=1}^{T} h_{it} \cdot I_{it} + \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{t=1}^{T} \sum_{m=1}^{M} c_{ij} \cdot T_{ijtm}$$

$$\text{subject to } (7) - (14), (21), (22)$$

$$I_{it}, X_{it}, \mu^s_{it} \geq 0; \alpha_{itm}, T_{ijtm} \in \{0,1\} \qquad\qquad i,j,t,m$$

This is a nonlinear model that can be easily linearized. We show the underlying linearization of (21) and (22) in Appendix A.

Example continued. Considering again the example given in Section 2.2, Figure 6 depicts the optimal solution obtained by the lot-streaming version of the model with total cost of 7. The values of the decision variables that are not 0 are indicated.

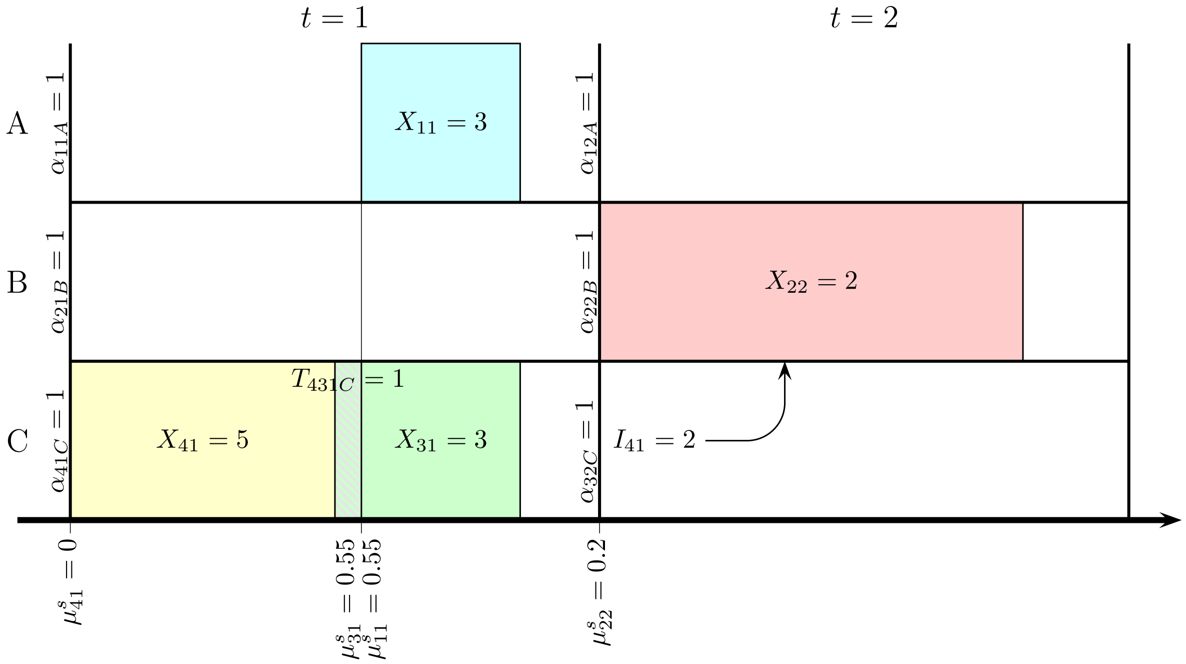


Figure 6: Illustration of decision variables for the lot-streaming case

Considering the inventory level of item 4 at the end of period 1, the new constraints (21) and (22) read in period 2 as follows.

$$\underset{I_{41}}{2} \geq \underset{a_{43}}{1} \cdot \min\left(\underset{X_{32}}{0}, \underset{\frac{1}{p_3}}{10} \cdot \left(\underset{\mu^s_{42}}{0} - \underset{\mu^s_{32}}{0}\right)\right) + \underset{a_{42}}{1} \cdot \min\left(\underset{X_{22}}{2}, \underset{\frac{1}{p_2}}{2.5} \cdot \left(\underset{\mu^s_{42}}{0} - \underset{\mu^s_{22}}{0}\right)\right) = 0$$

$$\underset{I_{41}}{2} + \min\left(\underset{X_{42}}{0}, \underset{\frac{1}{p_4}}{10} \cdot \left(\underset{\mu^s_{22}}{0} + \underset{p_2 \cdot X_{22}}{0.8} - \underset{\mu^s_{42}}{0}\right)\right) = 2 \geq$$

$$\underset{a_{43}}{1} \cdot \min\left(\underset{X_{32}}{0}, \underset{\frac{1}{p_3}}{10} \cdot \left(\underset{\mu^s_{22}}{0} + \underset{p_2 \cdot X_{22}}{0.8} - \underset{\mu^s_{32}}{0}\right)\right) + \underset{a_{42}}{1} \cdot \min\left(\underset{X_{22}}{2}, \underset{\frac{1}{p_2}}{2.5} \cdot \left(\underset{\mu^s_{22}}{0} + \underset{p_2 \cdot X_{22}}{0.8} - \underset{\mu^s_{22}}{0}\right)\right) = 2$$

Hence, (22) ensures that there is enough inventory for item 4 at the beginning of period 2. □

We prove in the following theorem that F_B is a special case of F_L and, as such, the optimal value of F_L is at least as good as F_B's.

Theorem 2. *We have $v(F_B) \geq v(F_L)$.*

Proof. We demonstrate the statement by showing that adding constraints (16) and (17) to F_L constraints (21) and (22) become redundant and can be dropped.

Consider $S(i) = \{j \in \Gamma(i) : \mu^s_{it} \geq \mu^s_{jt}\}$. Constraints (21) are then equivalent to $I_{i(t-1)} \geq \sum_{j \in S(i)} a_{ij} \min\left(X_{jt}, \frac{1}{p_j}\left(\mu^s_{it} - \mu^s_{jt}\right)\right)$. If $j \in S(i)$, then (15) imposes $W_{ijt} = 0$ and $\hat{X}_{ijt} \geq X_{jt}$. Using (16) and (17), we obtain

$$I_{i(t-1)} \geq \sum_{j \in \Gamma(i)} a_{ij}\hat{X}_{ijt} \geq \sum_{j \in S(i)} a_{ij} \cdot X_{jt} \geq \sum_{j \in S(i)} a_{ij} \min\left(X_{jt}, \frac{1}{p_j}\left(\mu^s_{it} - \mu^s_{jt}\right)\right) \quad i, t,$$

making constraint (21) redundant.

Let $Q(i,k)$, $k \in \Gamma(i)$ denote $Q(i,k) = \{j : j \in \Gamma(i) \text{ and } \mu^s_{kt} + p_k X_{kt} \geq \mu^s_{jt}\}$. The right hand-side of (22) can then be simplified to

$$\sum_{j \in Q(i,k)} a_{ij} \min\left(X_{jt}, \frac{1}{p_j}\left(\mu^s_{kt} + p_k X_{kt} - \mu^s_{jt}\right)\right).$$

Let us now distinguish two cases: $\mu^s_{kt} + p_k X_{kt} \leq \mu^s_{it}$ or $\mu^s_{kt} + p_k X_{kt} > \mu^s_{it}$.

We start by $\mu^s_{kt} + p_k X_{kt} \leq \mu^s_{it}$. Since $\mu^s_{kt} \leq \mu^s_{it} + p_i \cdot X_{it}$, from (15) clearly $W_{ikt} = 0$. Moreover, $W_{ijt} = 0$ for every j in $Q(i,k)$, which implies $\hat{X}_{ijt} \geq X_{jt}$. Thus,

$$I_{i(t-1)} \geq \sum_{j \in Q(i,k)} a_{ij} \cdot X_{jt} \geq \sum_{j \in Q(i,k)} a_{ij} \min\left(X_{jt}, \frac{1}{p_j}\left(\mu^s_{kt} + p_k X_{kt} - \mu^s_{jt}\right)\right) \quad i, k \in \Gamma(i), t.$$

Let now $\mu^s_{kt} + p_k X_{kt} > \mu^s_{it}$. We have to further consider two different scenarios.

If $\mu^s_{kt} + p_k X_{kt} < \mu^s_{it} + p_i X_{it}$, then similarly to the above case, it follows that $W_{ikt} = 0$ and $W_{ijk} = 0$ for every j in $Q(i,k)$. In terms it yields

$$\begin{aligned} I_{i(t-1)} + \min\left(X_{it}, \frac{1}{p_i}\left(\mu^s_{kt} + p_k X_{kt} - \mu^s_{it}\right)\right) \geq I_{i(t-1)} \geq \\ \geq \sum_{j \in Q(i,k)} a_{ij} \cdot X_{jt} \geq \sum_{j \in Q(i,k)} a_{ij} \min\left(X_{jt}, \frac{1}{p_j}\left(\mu^s_{kt} + p_k X_{kt} - \mu^s_{jt}\right)\right), \end{aligned}$$

making (22) redundant.

Finally let $\mu^s_{kt} + p_k X_{kt} \geq \mu^s_{it} + p_i X_{it}$. In this case (22) reads

$$\begin{aligned} I_{i(t-1)} + \min\left(X_{it}, \frac{1}{p_i}\left(\mu^s_{kt} + p_k X_{kt} - \mu^s_{it}\right)\right) = I_{i(t-1)} + X_{it} \geq \\ \geq \sum_{j \in Q(i,k)} a_{ij} \min\left(X_{jt}, \frac{1}{p_j}\left(\mu^s_{kt} + p_k X_{kt} - \mu^s_{jt}\right)\right). \end{aligned}$$

It is easy to see that this condition is dominated by $I_{i(t-1)} + X_{it} \geq \sum_{j \in \Gamma(i)} a_{ij} \cdot X_{jt}$, which is derived from (7) given that I_{it} and E_{it} are non-negative. □

3.3 Hybrid case

An interesting extension is a combination of the batching and lot-streaming cases, i.e. the raw materials might be delivered continuously to the production process of item i, but the newly produced items are only available if the whole batch is finished. In such a case we have to ensure that the whole production of successors can be supplied from stock of item i until the production is finished. To that end, we replace (22) by

$$I_{i(t-1)} \geq \sum_{j \in \Gamma(i)} a_{ij} \min\left(X_{jt}, p_j \left(\mu_{it}^s + p_i X_{it} - \mu_{jt}^s\right)^+\right) \quad i, t. \tag{23}$$

In (23) the start time μ_{it}^s of the production of item i was replaced by the finish time $\mu_{it}^s + p_i X_{it}$.

Both versions, the batching and lot-streaming cases, represent extreme situations, especially the continuous production case where it is possible to produce subsequent items in the BOM at the same time. Nevertheless, the batching scenario is more often found in practice. If we want to model a situation where it is not necessary to wait for the whole batch to be finished before items can be retrieved and used for further production steps, we might introduce a minimum production lead time τ_i and substitute in constraints (21) and (22) the starting time of item i with the term $\mu_{it}^s + \tau_i$.

3.4 Limitations

One assumption of big-bucket models is that at most one batch of each product is allows per time period. This assumption still holds for the presented models. Let us now consider the example from Section 2.2 with modified demand for item 2, $d_{21} = 2$ and $d_{22} = 0$. The only feasible solution to this instance is shown in Figure 7. Such a solution enforces two batches of item 4 in period 1 on machine C, which is not a feasible solution to our model. Belvaux and Wolsey (2001) develop a lot-sizing model, which accounts for multiple setups of the same item. In their approach the production sequence is not unique anymore. Adapting it to our case would require to compute starting times for each batch and thus blowing up the model size. Fandel and Stammen-Hegene (2006) develop a general lot-sizing and scheduling approach for that problem but they end up with a highly complex non-linear model. Experiences with such model formulations in similar settings show that the computational tractability of these kind of models is worse (see e.g. Almeder and Almada-Lobo, 2011).

4 Computational Comparison against MLCLSP

In this section we analyze the solution quality obtained by the traditional MLCLSP model and compare it with the new extended model formulation. All models were implemented

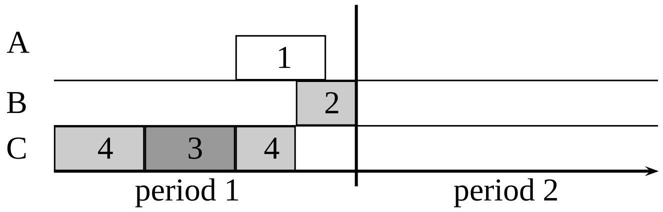


Figure 7: Example from Section 2.2 with changed demand for item 2.

in C++ using CPLEX 12.4 with Concert technology on a Intel Xenon E5540 CPU (using a single thread) and 4 GB of random access memory. Furthermore, we used the well-known (l, S) valid inequalities adapted by Clark and Armentano (1995) to the multi-level case. We add them dynamically within the branch-and-cut framework of CPLEX.

4.1 MLCLSP with Zero Lead Time and no Setup Carryover

Considering that the proposed model incorporates more details than MLCLSP, we first evaluate if solutions derived from the traditional MLCLSP model would also be optimal or feasible for our approach. That means we are testing if the lot sizes obtained by MLCLSP (without scheduling) can be scheduled within the given capacity. For this purpose we considered 4 classes of instances proposed by Tempelmeier and Derstroff (1996). Classes A and B are small instances with 10 items produced on 3 machines and a planning horizon of 4 periods. Class A instances consider setup cost but no setup times, whereas class B instances include also setup times. These classes are small enough that MLCLSP with zero lead time and no setup carryover can be solved easily with CPLEX within a second. Classes C and D are larger instances (40 items, 6 machines, 16 periods). For most of these instances optimal solutions of MLCLSP are unknown. We take the best known solutions for these classes obtained by a heuristic method proposed in Almeder (2010). All classes contain instances with a general and assembly product structure, with different demand profiles and different levels of capacity utilization.

Classical MLCLSP does not consider setup carryover. To avoid setup carryover in our formulation we introduce additional items (one for each machine) with no demand and force that setup carryovers are only possible for these additional items. The setup times and costs into these items are set to zero. The production quantities obtained by solving MLCLSP are fixed in our formulation (6)-(18) such that we only determine the production sequences, and start and finish times. We let CPLEX run for at most 600 seconds and classify the outcomes in three different categories:

Infeasible: CPLEX reports that the problem is infeasible, i.e. it is not possible to find a capacity feasible solution with the given production amount respecting synchronization conditions,

Feasible: CPLEX is able to find a capacity feasible solution with the given production quantities, which considers synchronization,

Unknown: CPLEX is neither able to generate a feasible solution, nor to prove infeasibility.

Table 3 shows the results of these tests. The values in brackets correspond to the number of tested solutions for each of the classes of test instances. We clearly observe that the vast majority of the solutions are infeasible. This further solidifies the need of our work. Since the batching constraint is more restrictive, fewer solutions are feasible under this assumption. But even without considering batching, which implies that predecessor and successor can be produced simultaneously, more than 85% of the instances are infeasible. It seems that the zero lead-time assumption leads to solutions that cannot be implemented in practice. On the other hand, having only a few periods, a 1-period lead time is also not meaningful.

Table 3: Checking the feasibility of solutions of the classical MLCLSP model.

	batching assumption			lot-streaming assumption		
class	infeasible	feasible	unknown	infeasible	feasible	unknown
A (1500)	99.33%	0.67%	0.00%	89.47%	10.53%	0.00%
B (600)	99.33%	0.67%	0.00%	87.67%	12.33%	0.00%
C (599)	99.83%	0.00%	0.17%	92.49%	0.00%	7.51%
D (573)	98.78%	0.00%	1.22%	84.99%	0.00%	15.01%

In Table 4 we aggregate the results of the different classes according to various criteria. The value in brackets corresponds to the number of tested solutions in each group. Differentiating by the product structure (general and assembly structure) and by the way items are assigned to machines (cyclic - the same machine is used for items on different levels; non-cyclic - only items at the same level or adjacent levels in the BOM are assigned to the same machines), we observe that most of the feasible solutions are obtained for problems with a general product structure and a non-cyclic machine-item relation (for the lot-streaming assumption). Furthermore, it is obvious that capacity-tight problems are more likely to be infeasible, because free capacity usually means more flexibility for the scheduling problem. The capacity utilization is defined by the total capacity requirement for production and setup for each machine divided by the total available capacity of that machine. Also, the setup cost level has an impact on the feasibility of the solution. High setup costs tend to generate solutions with large batches, which are more difficult to schedule than smaller batches.

Increasing capacity

A work-around which is often proposed to avoid infeasible plans to reserve some extra capacity for the scheduling step. So MLCLSP is first applied to a problem with reduced

Table 4: Breakdown with respect to BOM structure, capacity utilization, and setup cost.

	batching assumption			lot-streaming assumption		
group	infeasible	feasible	unknown	infeasible	feasible	unknown
general-noncyclic (823)	99.64%	0.00%	0.36%	77.16%	16.77%	6.08%
general-cyclic (814)	99.14%	0.86%	0.00%	96.68%	1.47%	1.84%
assembly-noncyclic (811)	99.63%	0.00%	0.37%	95.07%	0.86%	4.07%
assembly-cyclic (824)	98.91%	0.85%	0.24%	86.89%	9.10%	4.00%
capacity util. 90% (641)	100.00%	0.00%	0.00%	99.69%	0.00%	0.31%
capacity util. 70% (657)	100.00%	0.00%	0.00%	88.74%	7.15%	4.11%
capacity util. 50% (660)	96.82%	2.12%	1.06%	79.09%	14.85%	6.06%
setup cost - low (653)	96.63%	2.14%	1.23%	82.08%	9.65%	8.27%
setup cost - med. (656)	100.00%	0.00%	0.00%	90.24%	7.16%	2.59%
setup cost - high (656)	100.00%	0.00%	0.00%	92.07%	6.10%	1.83%

capacity hoping that afterwards lots can be scheduled such that capacity restrictions are not violated.

In order to investigate this approach we take again the solutions obtained by the classical MLCLSP model and try to solve the scheduling problem while increasing the capacity per period by 10%, 30%, and 50%. Table 5 reports the percentage of infeasible plans in the different groups. Percentages are given in ranges because a run time limit of 600 seconds was applied. For some of the problems no feasible solution was found within this time limit, as well as no proof of infeasibility could be obtained. But already the lower bounds on the number of infeasible instances are sufficient to spot the problem. It shows clearly that for the case of batching even if the capacity is increased by 50% most of the MLCLSP solutions are infeasible even if the original problem had only a theoretical capacity utilization of 50%. For the lot-streaming case the results are better but still more than 20% of the production plans are infeasible when the capacity is increased by 50%. Similar to the previous results a cyclic machine assignment and low capacity utilization lead to less infeasible plans. Figure 8 shows the share of classical MLCLSP solutions which cannot be scheduled for different levels of capacity increase.

4.2 MLCLSP with Zero Lead Time and Setup Carryover

In addition to classical MLCLSP the proposed model considers setup carryover. We have applied the full model with synchronization to the test classes A and B in order to gain some insights about the computational behavior of the model. The results are compared to the solution of classical MLCLSP with setup carryover. The solution of that model is a lower bound for the solution of model (6)-(18).

Table 6 shows the results obtained by CPLEX after 600 seconds run time. The columns report the number of solutions solved to optimality, the number of feasible solutions, the number of infeasible solutions, and the number of solutions with an unknown status, i.e.

Table 5: Percentage of infeasible plans with and without increased capacity

	batching				lot-streaming			
group	orig. cap.	cap.+10%	cap.+30%	cap.+50%	orig. cap.	cap.+10%	cap.+30%	cap.+50%
general-noncyclic (823)	99.6-100%	98.8-99.1%	98.1-99.0%	93.9-95.6%	77.2-83.2%	52.6-64.5%	18.1-41.1%	6.9-36.5%
general-cyclic (814)	99.1%	99.1%	96.3-96.9%	85.3-86.5%	96.7-98.5%	86.0-94.0%	49.3-70.1%	19.0-47.9%
assembly-noncyclic (811)	99.6-100%	99.3-100%	97.7-99.4%	95.6-98.5%	95.1-99.1%	91.7-98.5%	74.8-93.5%	49.6-75.5%
assembly-cyclic (824)	98.9-99.2%	98.5-99.2%	94.2-95.6%	82.0-85.2%	86.9-90.9%	68.6-76.6%	32.3-49.9%	16.0-38.3%
capacity util. 90% (641)	100%	100%	100%	99.5-100%	99.7-100%	94.9-96.9%	65.8-77.1%	39.0-59.9%
capacity util. 70% (657)	100%	100%	98.3-98.9%	91.8-93.9%	88.7-92.8%	76.9-86.3%	46.9-67.7%	18.1-47.2%
capacity util. 50% (660)	96.8-97.9%	95.2-96.8%	88.8-91.7%	76.5-80.3%	79.1-85.2%	64.4-76.5%	31.4-54.2%	16.1-43.6%
setup cost - low (653)	96.6-97.9%	94.6-96.8%	86.5-91.7%	71.1-79.3%	82.1-90.4%	68.8-80.4%	40.0-60.6%	23.0-48.7%
setup cost - med. (656)	100%	100%	99.7-99.8%	93.8-94.4%	90.2-92.8%	75.5-83.5%	42.4-63.4%	22.0-49.8%
setup cost - high (656)	100%	100%	99.8%	97.7-97.9%	92.1-93.9%	78.2-85.7%	46.6-67.2%	25.3-49.8%
all (3272)	**99.3-99.5%**	**98.9-99.3%**	**96.5-97.7%**	**89.2-91.5%**	**88.9-92.9%**	**74.6-83.3%**	**43.5-63.5%**	**22.8-89.5%**

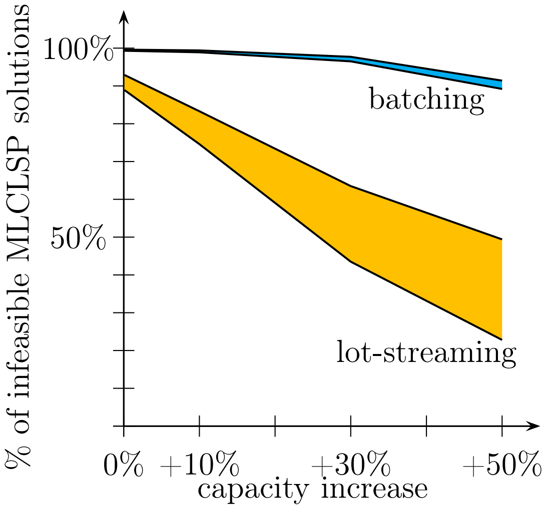


Figure 8: Percentage of infeasible MLCLSP solutions

we were neither able to find a feasible solution, nor could we prove infeasibility. We also report the average gap reported by CPLEX for all feasible solutions. The last column shows the average increase of the total costs of an optimal solution compared with the solution of classical MLCLSP with setup carryover but no synchronization.

Table 6: Results for our model formulation.

class	optimal	feasible	infeasible	unknown	avg. gap	cost incr.
			batching assumption			
A (1500)	177 (11.8%)	185 (12.3%)	889 (59.3%)	249 (16.6%)	11.6%	32.7%
B (600)	89 (14.8%)	88 (14.7%)	317 (52.8%)	106 (17.7%)	11.2%	28.5%
			lot-streaming assumption			
A (1500)	169 (11.3%)	902 (60.1%)	284 (18.9%)	145 (9.7%)	23.2%	5.9%
B (600)	79 (13.2%)	387 (64.5%)	76 (12.7%)	58 (9.7%)	19.8%	6.4%

The most important result of these tests is shown in the last column. It represents the additional cost of synchronization compared with classical MLCLSP, or, in other words, the underestimation of costs by MLCLSP compared to a more realistic and detailed model formulation.

A second outcome of these experiments is that there are quite a few instances with unknown status. Also the average gap for feasible solutions is still considerable which is due to the complexity of the new model formulation. Hence, future research should concentrate on better formulations and fast solution algorithms for this problem. Nevertheless, one main focus of this paper is to show the potential of integrated scheduling decisions within MLCLSP.

4.3 MLCLSP with One Period Lead Time and Setup Carryover

One way to overcome the feasibility problem of MLCLSP is to consider a minimum lead time of one period. As aforementioned this lead time may cause a substantial increase of inventory. Tempelmeier and Buschkühl (2009) used an extension of the class B test instances for MLCLSP with lead time and setup carryover. In order to guarantee feasibility they added two additional periods at the beginning with no external demand (in total now 6 periods). Since the problems have a three level product structure it is possible to satisfy an end-item demand in the third period. We will denote the new class of instances as B6.

We solve the test instances using classical MLCLSP without synchronization, but with one period lead time (as, for example, described by Tempelmeier and Buschkühl, 2009), as well as using our new model formulation with synchronization. We apply a 600 second run time limit to CPLEX. Table 7 compares the new solutions obtained by using the synchronization feature. The columns show the number of solutions solved to optimality, the number of feasible solutions, the number of instances where no solution has been obtained, the number of improved solutions and the average cost improvement (improvement of the upper bound) of the synchronization model compared to MLCLSP with lead time.

For the batching case we can solve 151 instances to optimality, for 448 instances we find a feasible solution and only for one instance we are not able to compute a feasible solution within the specified run time. For the lot-streaming case less instances are solved to optimality. But for all other instances we obtain feasible solutions.

The column labeled *UB impr.* denotes the average improvement of the objective function of the best solution found (optimal or feasible) compared to the optimal solution of MLCLSP with lead time. Considering lead time in a big-bucket model leads to a clear overestimate of the optimal costs of at least 30.6% for batching and 39.4% for lot-streaming due to the increased inventory levels carried from one-period to the next just to maintain the feasibility of the solution. We note that for the instances with low capacity utilization and high setup cost, the cost overestimation by the MLCLSP solution with one period lead time is higher than for the other cases.

5 Conclusions

Our extensive computational study clearly indicates that the solutions obtained by MLCLSP are either infeasible or lead to excessive work-in-process (from 15% to 60% increase in overall cost). The need for synchronization is thus clearly established. This work is the first step in such a direction to develop a synchronisation mechanism in a big-bucket lot-sizing framework. We present here two extensions of MLCLSP representing the two extreme cases of batching (i.e. a successor item can only be produced if the full batches of the predecessor items are finished) and lot-streaming (i.e. the successor item can be produced simultaneously

Table 7: Comparing MLCLSP with one period lead time with the new model with synchronization for the class B6 test instances.

	batching assumption					lot-streaming assumption				
group	opt.	feas.	unk.	improved	UB impr.	opt.	feas.	unk.	improved	UB impr.
general-noncyclic (150)	48	102	0	147 (98%)	33.2%	17	133	0	149 (99%)	32.7%
general-cyclic (150)	55	95	0	150 (100%)	29.4%	1	149	0	146 (97%)	29.9%
assembly-noncyclic (150)	7	142	1	141 (94%)	24.7%	3	147	0	150 (100%)	23.2%
assembly-cyclic (150)	41	109	0	150 (100%)	34.8%	6	144	0	150 (100%)	34.8%
capacity util. 60% (120)	2	117	1	108 (90%)	20.4%	0	120	0	117 (98%)	18.6%
capacity util 47% (120)	30	90	0	120 (100%)	29.8%	1	119	0	119 (99%)	29.9%
capacity util. 33% (120)	69	51	0	120 (100%)	41.5%	23	97	0	120 (100%)	41.5%
setup cost - low (120)	4	116	0	108 (90%)	20.4%	6	114	0	117 (98%)	18.6%
setup cost - med. (120)	13	90	0	120 (100%)	29.8%	6	114	0	119 (99%)	29.9%
setup cost - high (120)	67	51	1	120(100%)	41.5%	9	111	0	120 (99%)	41.5%
all (600)	**151**	**448**	**1**	**588 (98%)**	**30.6%**	**27**	**573**	**0**	**595 (99%)**	**39.4%**

with its predecessor items as long as it uses a different resource). It is straight forward to introduce any possible lead time in between these two extremes if technical restrictions are demanding it.

While commercial solvers are capable of solving small to mid size instances and many larger instances, large-scale cases need further investigations. Future work should be devoted to develop tighter model formulations and MIP-based heuristic solution approaches.

References

Almada-Lobo, B., Carravilla, M., Oliveira, J., 2008. A note on "The capacitated lot-sizing and scheduling problem with sequence-dependent setup costs and setup times. Computers & Operations Research 35, 1374–1376.

Almada-Lobo, B., Klabjan, D., Carravilla, M., Oliveira, J., 2007. Single machine multi-product capacitated lot sizing with sequence-dependent setups. International Journal of Production Research 45, 4873–4894.

Almeder, C., 2010. A hybrid optimization approach for multi-level capacitated lot-sizing problems. European Journal of Operational Research 200, 599–606.

Almeder, C., Almada-Lobo, B., 2011. Synchronization of scarce resources for a parallel machine lotsizing problem. International Journal of Production Research 49 (24), 7315–7335.

Belvaux, G., Wolsey, L., 2001. Modelling practical lot-sizing problems as mixed-integer programs. Management Science 47, 993–1007.

Berretta, R., Franca, P., Armentano, V., 2005. Metaheuristic approaches for the multilevel resource-constraint lot-sizing problem with setup and lead times. Asia - Pacific Journal of Operational Research 22 (2), 261–286.

Berretta, R., Rodrigues, L.F., 2004. A memetic algorithm for a multistage capacitated lot-sizing problem. International Journal of Production Economics 87, 67–81.

Billington, P., McClain, J., Thomas, L., 1983. Mathematical programming approach to capacity-constrained MRP systems: Review, formulation and problem reduction. Management Science 29, 1126–1141.

Bitran, G., Yanasse, H., 1982. Computational complexity of the capacitated lot size problem. Management Science 28, 1174–1186.

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, A., Armentano, V., 1995. The application of valid inequalities to the multi-stage lot-sizing problem. Computers & Operations Research 22, 669–680.

Clark, A., Almada-Lobo, B., Almeder, C. 2011. Lot Sizing and Scheduling: Industrial extensions and research opportunities. International Journal of Production Research 49, 2457–2461.

Drexl, A., Kimms, A., 1997. Lot sizing and scheduling - survey and extensions. European Journal of Operational Research 99, 221–235.

Fandel, G., Stammen-Hegene, C., 2006. Simultaneous lot sizing and scheduling for multi-product multi-level production. International Journal of Production Economics 194, 308–316.

Gómez Urrutia, E.D., Aggoune, R., Dauzère-Pérès, S., 2012. New Integrated Approach For Solving Multi-level Lot Sizing and Scheduling problems, in: 9th International Conference of Modeling, Optimization and Simulation - MOSIM´12.

Gupta, D., Magnusson, T., 2005. The capacitated lot-sizing and scheduling problem with sequence-dependent setup costs and setup times. Computers & Operations Research 32, 727–747.

Haase, K., 1996. Capacitated lot-sizing with sequence dependent setup costs. Operations Research Spektrum 18, 51–59.

Karimi-Nasab, M., Seyedhoseini, S.M., 2013. Multi-level lot sizing and job shop scheduling with compressible process times: A cutting plane approach. European Journal of Operational Research 231, 598–616.

Maes, J., McClainLuk, J., Van Wassenhove, N., 1991. Multilevel capacitated lotsizing complexity and LP-based heuristics. European Journal of Operational Research 53, 131–148.

Meyr, H., 2004. Simultane Losgrößen- und Reihenfolgeplanung bei mehrstufiger kontinuierlicher Fertigung. Zeitschrift für Betriebswirtschaft, 74, 585–610.

Pitakaso, R., Almeder, C., Doerner, K.F., Hartl, R.F., 2006. Combining population-based and exact methods for multi-level capacitated lot-sizing problems. International Journal of Production Research 44, 4755–4771.

Seeanner, F., Meyr, H., 2013. Multi-stage simultaneous lot-sizing and scheduling for flow line production. OR Spectrum 35, 33?73.

Stadtler, H., 2005. Multilevel capacitated lot-sizing and resource-constrained project scheduling: an integrated perspective. International Journal of Production Research 43, 5253–5270.

Stadtler, H., 2011. Multi-level single machine lot-sizing and scheduling with zero lead times. European Journal of Operational Research 209, 241–252.

Tempelmeier, H., Buschkühl, L., 2008. Dynamic multi-machine lotsizing and sequencing with simultaneous scheduling of a common setup resource. International Journal of Production Economics 113, 401–412.

Tempelmeier, H., Buschkühl, L., 2009. A heuristic for the dynamic multi-level capacitated lotsizing problem with linked lotsizes for general product structures. OR Spectrum 31, 385–404.

Tempelmeier, H., Derstroff, M., 1996. A Lagrangean-based heuristic for dynamic multilevel multi-item constrained lotsizing with setup times. Management Science 42, 738–757.

Toledo, C.F.M., de Oliveira, R.R.R., Morelato Franca, P., 2013. A hybrid multi-population genetic algorithm applied to solve the multi-level capacitated lot sizing problem with backlogging. Computers & Operations Research 40, 910–919.

Urrutia, E., Aggoune, R., Dauzère-Pérès, S. New integrated approach for solving multi-level lot sizing and scheduling problems. Proceedings of the 9th International Conference on Modeling, Optimization & Simulation.

Wagner, H., Within, T., 1958. Dynamic version of the economic lot size model. Management Science 5, 89–96.

Wolsey, L., 2002. Solving multi-item lot-sizing problems with an MIP solver using classification and reformulation. Management Science 48, 1587–1602.

Wu, T., Shi, L., Geunes, J., Akartunali, K., 2012. On the equivalence of strong formulations for capacitated multi-level lot sizing problems with setup times. Journal of Global Optimization 53, 615–639.

Wu, T., Shi, L., Geunes, J., Akartunali, K., 2011. An optimization framework for solving capacitated multi-level lot-sizing problems with backlogging. European Journal of Operational Research, 214 (2), 428–441.

Xie, J., Dong, J., 2002. Heuristic genetic algorithms for general capacitated lot-sizing problems. Computers and Mathematics with Applications 44, 263–276.

Zhu, X., Wilhelm, W., 2006. Scheduling and lot sizing with sequence-dependent setup: A literature review. IIE Transactions 38, 987–1007.