Rolling horizon formulations for short-to-medium term production planning

Luis Guimarães[a], Diego Klabjan[b], Bernardo Almada-Lobo[a]
[a] INESC TEC, Faculdade de Engenharia, Universidade do Porto, Porto, Portugal
[b] Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, Illinois, USA
guimaraes.luis@fe.up.pt, d-klabjan@northwestern.edu, almada.lobo@fe.up.pt

February 28, 2015

Abstract

In operational production planning several companies have to size and schedule production lots on a set of parallel machines to satisfy forecasted demand, while facing sequence dependent changeover times and costs. Motivated by a case study in a beverage company, we exploit the practice of rolling basis planning to develop efficient approaches to the problem. The horizon is decomposed in two parts: the initial periods explicitly consider the production sequences to obtain detail schedules, while in the remaining periods a rough plan is generated to give an estimation of future costs and capacity. Several modeling alternatives proposed in the literature are reviewed and a new formulation that includes the setup loss in the future periods based on the loss experienced in the detail part of the horizon is proposed. An important contribution is an innovative iterative method to improve the accuracy of the approximate parameters used in the context of the simplified models. We assess the performance of several alternatives by simulating the implementation of solutions on a rolling horizon by using instances generated based on the features arising in the beverage industry. The tests simulate the planning environment in practice by generating demand forecasts followed by the implementation of the production plans. The results show that applying the iterative approach on the approximate methods can generate solutions that better trade-off costs and stockouts.

1. Introduction

In this paper we discuss practical modeling techniques to integrate medium and short term decisions for production planning and scheduling in a single facility. We are motivated by a case study in a beverage company that produces mineral and sparkling water, beer and soft drinks in a set of different production centers. At each facility a series of filling lines, the production process bottleneck, is available to produce a wide range of final items. These resources are usually dedicated to produce a certain type of final products (e.g. kegs, glass bottles, cans), however, nowadays advanced manufacturing technology has allowed for the appearance of more flexible filling lines. This flexibility and the existence of parallel production resources forces simultaneous planning of all or at least most of the available filling lines in order to achieve more efficient plans.

Filling lines can only produce one product at a time, being adjusted to fill a given liquid, container type and size, and final package. Whenever a product changeover is necessary it may require several mechanical adjustments in the filling line and cleansing operations. The number and complexity of the operations triggered depend on the previous and following products, i.e. setup time and costs are dependent on the production sequence. Tackling accurately these sequence-dependent setups is vital to ensure the competitiveness of a company, as frequent and long setups can reduce substantially the operational efficiency of filling lines. Additionally, recent market dynamics led companies to increase the number of products and work with less stock, by delivering products more frequently, which further stresses the need for efficient plans due to the need for extra setups.

Production planning in the case study is a hierarchical process with several echelons, each containing different aims and planning horizons. In particular, medium term planning focuses on defining a plan for the next 6 to 8 weeks and serves as an input to the short term planning that defines the schedule of operations for the next week. Mid term planning decisions consider the sizing of filling line production lots and overtime utilization (whenever possible). Short term decisions look only at a filling line at a time and try to schedule the production lots defined in the previous level with the objective of minimizing the sequence-dependent setup times and tardinesses.

The company in question uses SAP APO, an advanced planning system (APS), to perform its supply chain planning tasks. APSs are modular planning software which extract data from traditional enterprise resource planning (ERP) systems and support decision making by using pre-defined mathematical models, heuristics and other quantitative techniques, before feeding back ERP with the final solutions for their execution (see Fleischmann and Meyr [2003]). APS modules' architecture is aligned with an hierarchical decision system and with the supply chain functions defined by the supply chain planning matrix (Fleischmann et al. [2008]). Specifically in our case study, the two planning levels discussed above are tackled in two different modules. The medium term decisions are made in the Supply Network Planning (SNP) module, while short term planning is carried out using the Production Planning and Detail Scheduling (PP/DS) module. Next, we describe the modules as installed in the company at the time of writing.

SNP delivers a weekly bucket-oriented plan to define the production lot sizes on the series of available filling lines. The algorithm in use is an adapted version of the Capable-To-Match (CTM) heuristic of SAP APO. It considers *a priori* a preferential filling line for every product to each the forecasted demand is assigned, usually the filling line with the fastest processing time. When the capacity of a filling line is exceeded at a given time period the allocated productions are moved according to pre-defined rules. These rules may imply moving production quantities to earlier periods or to another filling line capable of producing the product and with a capacity surplus. Overtime allocation is managed by exception. During the creation of the lot sizes the sequence-dependent setup times are not considered, being instead included in the average processing times used.

PP/DS receives the lots planned in SNP as an input and tries to sequence them while minimizing the sequence-dependent setup times. This step is performed by a dispatching rule, which sequences products by grouping them according to their features and treating time as continuous. The heuristic considers one filling line at a time and just the first week of the medium term planning horizon. The resulting detailed schedule is then sent to SAP ERP to be executed.

A planning step consists on the execution of SNP followed by PP/DS. The medium term plan is updated as new demand forecasts become available and the planning horizon is rolled forward one week. It is common for managers to manually change the SNP plans before sending them to PP/DS and changes to PP/DS are also made before the plans' implementation.

This type of an hierarchical approach has been proved to give not only suboptimal solutions in the presence of sequence-dependent setups, but also to pose challenges in terms of solution feasibility. In Pinedo and Kreipl [2004] a similar framework is developed for a brewery company, with the authors highlighting that "the results coming out of the detailed scheduling problem may be, for various reasons, not acceptable." On one hand, the average processing times in medium term planning may lead to overestimation of the capacity, which results in infeasible plans and forces medium term planning to be run again. On the other hand, the possible underestimation can result inefficiency plans. The same problem has been reported in Mateus et al. [2010] in an iterative approach to the integrated lot sizing and sequence-dependent setup scheduling problem. Therefore, it is imperative to tackle these two levels of decisions simultaneously to correctly evaluate capacity and determine the lot sizes. A model considering the lot sizing and sequencing decisions over the medium term horizon would address this problem.

However, such type of models are known to pose hard optimization challenges and the fact that only the first period or periods are actually implemented questions the reasoning of expanding the sequencing decisions over such a longer horizon. Furthermore, forecasts for more distant periods are likely to change once the horizon is rolled forward and managers often revise the plans to cope with possible machine breakdowns, stock-outs, last minute imperative orders, and other disruptive events. This re-planning can occur more than once a day leaving limited time for optimizing complex models and stressing the need for a computational efficient method of generating plans under these extreme conditions.

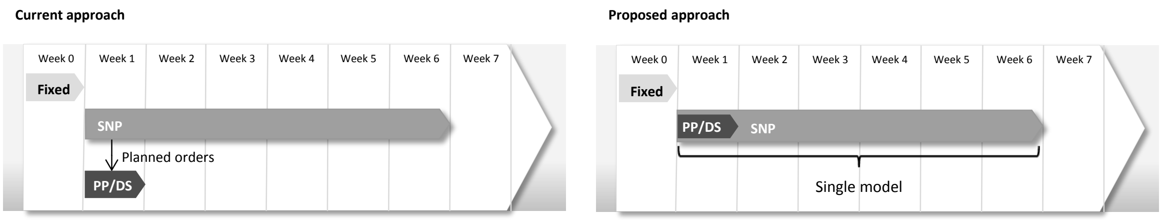

Figure 1: Medium and short term planning horizons

The objective of this paper is to respond to these requirements by developing new formulations and methods that link short and medium term production planning in a single facility/multi-machine environment, while being computationally efficient for their use under rolling planning and event-based re-planning.

To materialize our objectives the idea is to use approximate models that create highly detailed plans in the beginning of the planning horizon and make an estimation of future capacity utilization and costs in later periods. Such models are computationally less expensive than models integrating lot sizing and scheduling with sequencing over the entire medium term horizon enabling a fast generation and re-generation of plans as required in practice. Nonetheless, the quality of the solutions implemented in the detailed horizon is affected by the quality of the approximations made in the remaining horizon. We study the effect of different approximations and develop an iterative method aiming to improve the accuracy of the parameters used in the approximations. Figure 1 depicts the differences between the current approach followed by the company and the proposed approach.

To validate the new approach we simulate the planning process occurring in practice. First the demand forecasts are generated using a time series based method, next these forecasts are then used in the approximate model to create the production plans. The final step relates to the plans' implementation and the cost evaluation considering the actual demand observed including the evaluation of stockouts introduced by forecasting errors.

Our contributions are as follows. We develop new formulations to use in rolling planning considering lot sizing and scheduling with sequencing decisions. These new formulations use a lot sizing and scheduling model with sequencing decisions that do not require the estimation of a maximum number of setups as the ones previously found in the literature. We also introduce a new approximate model that uses the information on setup time experienced in detailed time periods to reduce the capacity in the simplified horizon. On the methodology front, an important contribution is the iterative method used to improve the accuracy of the parameters defined in the context of approximate models. The method is modular and can be applied in several distinct model formulations as shown herein.

In the remainder of the paper we start by reviewing, in Section 2, the most important literature in the context of the current work. In Section 3 the complete lot sizing and scheduling model with sequencing decisions linking short and medium term is introduced. Based on this model, Section 4 presents the approximate models to be used in rolling planning and Section 5 introduces the new iterative method to estimate the parameters required in the approximate models. In Section 6, the several alternatives discussed are tested by simulating the industry planning process. We finish by summarizing conclusions from the conducted work and by setting potential future work.

2. Literature review

Much of the recent research dedicated to the integration of lot sizing and scheduling problems has focused on improving the plans' detail. This has been shown by several authors (Gopalakrishnan et al. [2001], Porkka et al. [2003], Haase [1996]) to improve the utilization of the capacity and reduce costs. Many real-world problems inspired model extensions such as: parallel machines, setup carry-over, sequence-dependent setup times and costs, setup crossover, among others. On one hand, these extensions turn the mathematical models more realistic and increase their potential applicability by allowing for better and more realistic plans. On the other hand, models become much more complex due to the expansion in the number of variables and constraints to capture these features, leading to hard optimization challenges and solution times that may not be aligned with business practice. Formulations integrating sequencing decisions in lot sizing have been improved over the recent years and many solution approaches have been proposed. Among the papers introducing solution approaches we highlight Kang et al. [1999], Meyr [2002], Kovács et al. [2009], James and Almada-Lobo [2011], Guimarães et al. [2013b]. Despite these improvements, problems of relevant size to real practice still pose difficulties.

As discussed before, business practice establishes that these improved models and solutions are likely to be applied in a rolling horizon basis. Only the first part of the plan is actually implemented, corresponding to the initial time periods. The remaining part serves the purposes of estimating future costs and capacity shortages, in order to account for their impact on the nearer decisions. Longer forecast horizons are also necessary in the presence of products that are only seldom produced within a year. Having this premise in mind, some authors have incorporated these principles behind rolling planning on lot sizing and scheduling using two distinct approaches. One explores the idea of an *internal* rolling scheduling using an implicit time decomposition of rolling planning to be able to efficiently handle large instances. The other focuses on the *external* rolling horizon defined by the successive planning steps to develop efficient mathematical formulations that trade-off the plans detail and the computational effort, see Figure 2. These two approaches share a common time structure partitioning the planning horizon into three parts: fixed, detailed and simplified horizons. The fixed horizon is associated with previous iterations of the method or previous planning steps. The detailed horizon embeds at least all the time periods to be implemented in the next iteration and uses an accurate model to express the problem. The simplified horizon uses an approximation of the exact model. Figure 2 depicts the difference between the horizons in the two approaches.

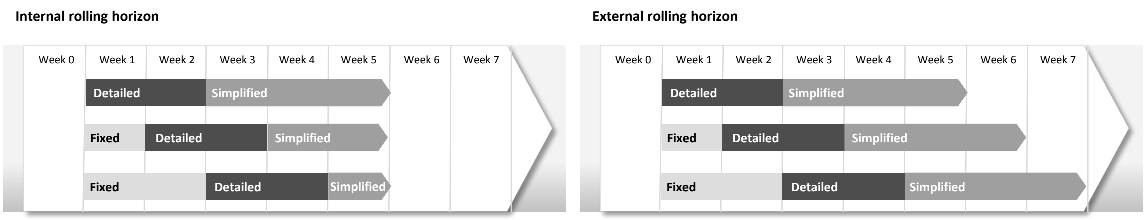

Figure 2: Internal and external rolling horizons

Table 1: Summury of related research

Reference	Planning environment			Modelling		Rolling horizon type
	General*	Setups	Feasibility	Objective Function**	Simplification Strategy	
Clark and Clark [2000]	SL, PM	Seq-Dep	Backlog	H, B	Increase processing times	Internal and External
Clark [2003]	ML, PM	Seq-Dep	Backlog	H, B	Increase processing times	Internal
Stadtler [2003]	ML, MR	Seq-Ind	Overtime	H, S, O	Capacity Reduction	Internal
Clark [2005]	SL, SM	Seq-Ind	Backlog	H, B	Increase processing times / Capacity reduction	Internal and External
Araujo et al. [2007]	SL, SM	Seq-Dep	Backlog	H, B, S	Sequence Independent Setups	Internal and External
Tiacci and Saetta [2012]	SL, SM	Seq-Dep	-	H, S	Sequence Independent Setups	External
This paper	SL, PM	Seq-Dep	Overtime	H, S, O	Increase processing times / Capacity reduction / Sequence Independent Setups	External

* SL - Single level, ML - Multi level, SM - Single machine, PM - Parallel machines, MR - Multi resource
** H - Holding costs, B - Backlog costs, S - Setup costs, O - Overtime costs

Table 1 summarizes the main features of the most relevant work in the context of the current paper with all models considering a single facility. Several criteria are used to classify the papers, namely the production environment, the types of setups, the strategy considered to address feasibility issues of the plans, components of objective functions, the simplification strategy used in the simplified horizon and the type of the rolling horizon approach.

Clark and Clark [2000] propose a mathematical formulation to be used in the external rolling horizon approach to solve a parallel machine lot sizing and scheduling problem with sequencing decisions. Their idea is to ignore binary setup decisions in the simplified horizon, either by relaxation or through an increase in the values of the processing times to incorporate the loss in setup times. Updating the processing times is non-trivial due to the simultaneous effect of the sequence-dependent setup times and lot sizes. Different approximations are proposed and compared by using a static instance as well as by means of a simulation of external rolling horizon planning. Despite tackling a different problem, a subsequent paper (Clark [2003]) revises the method for estimating the corrected processing times by building an MIP model to estimate the increase. These approximations are also combined with internal rolling scheduling based on a modified version of the relax-and-fix heuristic to solve the yet difficult detailed scheduling. However, their performance was not tested in the external rolling horizon environment. Clark [2005] further extends the approximation method in the rolling heuristics and conducts a thorough computational study using instances both with perfectly-known demand and including forecasting errors during rolling planning. Besides suggesting new methods to update the processing times, the author also investigates approximation schemes in which the processing times are kept, but capacity is reduced accordingly to the estimated loss in the setup times. In contrast to the two previous works and this paper, the setup times incurred are not dependent on the sequence of production.

An internal rolling horizon approach is designed in Stadtler [2003] for the multilevel lot sizing problem with setup times and multiple constrained resources. Their idea is to limit the number of detailed periods to be able to use a tighter model formulation. The detailed window is then deployed (or partially deployed) and internally rolled forward until a complete solution is available for the original planning horizon.

The general lot sizing and scheduling problem (GLSP) is modified in Araujo et al. [2007] in the spirit of rolling planning. The idea is to schedule products in the simplified horizon, but ignoring both setup times and cost. The authors also develop local search heuristics to solve the problem of the detailed horizon with realistic data within acceptable computational times. Later, Tiacci and Saetta [2012] extended this approach by considering sequence-independent setup costs in the simplified horizon and by proposing a formula for their estimation. The approach is validated by conducting experiments on data generated based on a case study of a company from the wood floor sector.

The formulations presented in this paper are, to the best of our knowledge, the first that apply rolling horizon principles using lot sizing and scheduling models which do not require the estimation of a maximum number of setups per period. We also propose an approximation model that reduces future capacity based on the setups observed in detailed time periods. A distinguishing element of our work is the idea to improve the accuracy of the parameters by incorporating additional information into the estimations on an iterative basis.

3. The complete lot sizing and scheduling model

The problem that considers the integration of sequencing decisions in the lot sizing and scheduling problem is known in the literature as the CLSD, which is an extension of the original Capacitated Lot Sizing Problem (CLSP). The objective is to minimize the total expenditure in inventory, setup and overtime costs over a finite planning horizon T. A plan simultaneously defines for every time period the production quantities and sequences for N products on a set of parallel capacitated machines. Such problem tackles SNP and PP/DS decisions all together. Like in the APO modules, demand is assumed to be known from forecasts and is to be met without backlog, which is a common setting in the industry. Overtime can be used to face the potential capacity shortages. In the context of this work, overtime refers to the use of additional days or shifts that are still available, e.g. non-working days (Saturdays, Sundays or holidays) or a third shift in a production line operating on a two shift schedule. Hence, both the extra capacity and costs assume discrete values as opposed to most literature that considers these decisions as continuous (Özdamar and Birbil [1998], Stadtler [2003]). Since production lines have dedicated crews it is possible to schedule extra time independently.

Sequencing decisions are introduced since both the setup times and costs are dependent on the production sequence. In production lines the setup state is preserved over adjacent periods and also over idle periods of the machines. For operational reasons changeovers are not allowed to overlap between periods forcing them to start and end within a single time period. Due to technological constraints each production line can only produce a subset of all products.

To formulate the CLSD problem with parallel machines and overtime decisions, hereafter called (CSP), consider the following parameters.

Sets and indices

- i, j — products, $i, j = 1, \ldots, N$.
- t — time periods, $t = 1, \ldots, T$.
- m — machines (production lines), $m = 1, \ldots, M$.
- o — overtime types, $o = 1, \ldots, O$. In example, $o = 1$ can correspond to Saturdays and $o = 2$ to Sundays.
- $\mathscr{A}_m$ — set of products that can be produced on machine m.

Data

d_{it}	demand of product i in period t (units)
h_{it}	holding cost of one unit of product i in period t
q_{imt}	maximum number of production lots of product i on machine m in period t
r^o_{mt}	cost of overtime type o on machine m in period t
cap_{mt}	normal capacity of machine m in period t (time)
cap^o_{mt}	overtime capacity of type o on machine m in period t (time)
p_{im}	processing time of product i on machine m
b_{imt}	upper bound on production quantity of product i on machine m in period t
st_{ijm}	time required to perform a changeover from product i to product j on machine m
sc_{ijm}	cost incurred when performing a changeover from product i to product j on machine m

The decision variables to be optimized are:

I_{it}	stock of product i at the end of period t
B^o_{mt}	(=1) if overtime capacity type o is used on machine m in period t
X_{imt}	quantity of product i to be produced on machine m in period t
Z_{imt}	(=1) if machine m is set up for product i at the beginning of period t
T_{ijmt}	number of changeovers from product i to product j performed on machine m in period t.

The mixed integer mathematical formulation (MIP) for the CSP reads:

$$\text{CSP} \quad \min \sum_{i,t} h_{it} \cdot I_{it} + \sum_{m,t,i,j \in \mathscr{A}_m} sc_{ijm} \cdot T_{ijmt} + \sum_{o,m,t} r^o_{mt} \cdot B^o_{mt} \tag{1}$$

$$\text{s.t.} \quad I_{i,t-1} + \sum_{m | i \in \mathscr{A}_m} X_{imt} = d_{it} + I_{it} \qquad \forall i, t, \tag{2}$$

$$\sum_{i \in \mathscr{A}_m} p_{im} \cdot X_{imt} + \sum_{i,j \in \mathscr{A}_m} st_{ijm} \cdot T_{ijmt} \leq cap_{mt} + \sum_o B^o_{mt} \cdot cap^o_{mt} \qquad \forall m, t, \tag{3}$$

$$X_{imt} \leq b_{imt} \cdot \left(\sum_{j \in \mathscr{A}_m} T_{jimt} + Z_{imt} \right) \qquad \forall m, i \in \mathscr{A}_m, t, \tag{4}$$

$$\sum_{i \in \mathscr{A}_m} Z_{imt} = 1 \qquad \forall m, t, \tag{5}$$

$$Z_{imt} + \sum_{j \in \mathscr{A}_m} T_{jimt} = \sum_{j \in \mathscr{A}_m} T_{ijmt} + Z_{im,t+1} \qquad \forall m, i \in \mathscr{A}_m, t, \tag{6}$$

$$\{(i,j) : T_{ijmt} > 0\} \text{ does not include disconnected subtours} \qquad \forall m, t. \tag{7}$$

$$X, I \geq 0, \quad Z \in \{0,1\}, \quad T_{jimt} \in \{0, \ldots, q_{imt}\}, \quad B \in \{0,1\}. \tag{8}$$

The objective function (1) minimizes the sum of holding, setup and overtime costs. We assume that production costs on the different machines are fixed and time independent. The demand balancing constraints are described by (2). Production time plus the time lost in setup operations should not exceed the available normal capacity incremented by the overtime decisions on each machine (3). Constraints (4) link production quantities with the machine setup state: production may only occur if a setup is carried over from the previous period or at least one setup is performed in the period. Constraints (5) ensure that the machine is set up for a single product in the beginning of each time period, while (6) keep trace of each machine configuration balancing the flow of setups as follows. If there are no setups in period t the machine configuration is carried to period $t+1$. On the other hand, for each product i three cases may appear: (i) more input than output setups, (ii) more output than input setups and (iii) equal number of input and output setups. In the first case the machine has to be set up for product i in the beginning of the next period $t+1$ ($Z_{im,t+1} = 1$). The opposite scenario, the second case, forces a setup for product i to be carried over from the previous period ($Z_{imt} = 1$). The third case happens when the product is neither the first nor the last in the sequence, or it is not part of the production sequence of the machine in the period.

Constraints (7) prevent disconnected subtours, i.e. sequences that start and end at the same setup state. Guimarães et al. [2013a] have shown in their study that model efficiency is directly linked to the selection of the proper subtour elimination constraints. We choose to use single commodity flow type constraints since as suggested by the results of this study, they yield a very computational efficient model which is able to provide feasible integer solutions even for large instances of the single machine variant.

Let us introduce the binary setup state variables Y_{imt} which equal one, if machine m is prepared to produce product i in period t, or zero otherwise. We consider the machine to be prepared for product i, in case either a setup in the period exists or a setup is carried over from a previous period. The following constraints (9)-(10) link variables Y_{imt} to both Z_{imt} and T_{ijmt}, while (11) replace (4) to give a tighter formulation.

$$\sum_{j \in \mathscr{A}_m} T_{jimt} + Z_{imt} \geq Y_{imt} \qquad \forall m, i \in \mathscr{A}_m, t, \tag{9}$$

$$\sum_{j \in \mathscr{A}_m} T_{jimt} + Z_{imt} \leq q_{imt} \cdot Y_{imt} \qquad \forall m, i \in \mathscr{A}_m, t, \tag{10}$$

$$X_{imt} \leq b_{imt} \cdot Y_{imt} \qquad \forall m, i \in \mathscr{A}_m, t. \tag{11}$$

Additionally, we also introduce the continuous flow variables F_{ijmt}, which capture the commodity flow from product (node) i to product (node) j on machine m in period t. In order to model setup carry-over an artificial node indexed by 0 is introduced which acts as the source of the flow. This formulation prevents disconnected subtours by imposing the connectivity of the graph induced by the setups selected through the changeover variables (T_{ijmt}). It forces the existence of a path connecting the artificial node to each one of the products in the sequence (unitary Y_{imt}). The constraints (7) are then defined as below:

$$\sum_{j \in \mathscr{A}_m} F_{0jmt} = \sum_{j \in \mathscr{A}_m} Y_{jmt} \qquad \forall m, t, \tag{12}$$

$$\sum_{j \in \mathscr{A}_m \cup \{0\}} F_{jimt} - \sum_{j \in \mathscr{A}_m} F_{ijmt} = Y_{imt} \qquad \forall m, i \in \mathscr{A}_m, t, \tag{13}$$

$$F_{0imt} \leq |\mathscr{A}_m| \cdot Z_{imt} \qquad \forall m, i \in \mathscr{A}_m, t, \tag{14}$$

$$F_{ijmt} \leq |\mathscr{A}_m| \cdot T_{ijmt} \qquad \forall m, i \in \mathscr{A}_m, j, t. \tag{15}$$

The amount of commodity flow forced to leave the source is defined by the number of paths needed, which is equivalent to the number of products produced in the time period imposed by (12). The flow balance constraints are expressed by (13) which ensure that a unitary flow is sent to every selected node, corresponding to a path from the source to every product being produced in the time period. A positive setup state acts as a unitary demand of the flow. Both (14) and (15) impose an upper bound on the amount of flow traversing the arcs. By (14), the flow can only leave the source to the first product in the sequence, while (15) guarantees that the flow only transverses arcs in the current solution.

As shown in Guimarães et al. [2013a], although the model has a good performance on large instances, it is evident that in case the number of products and/or machines increases, its performance is

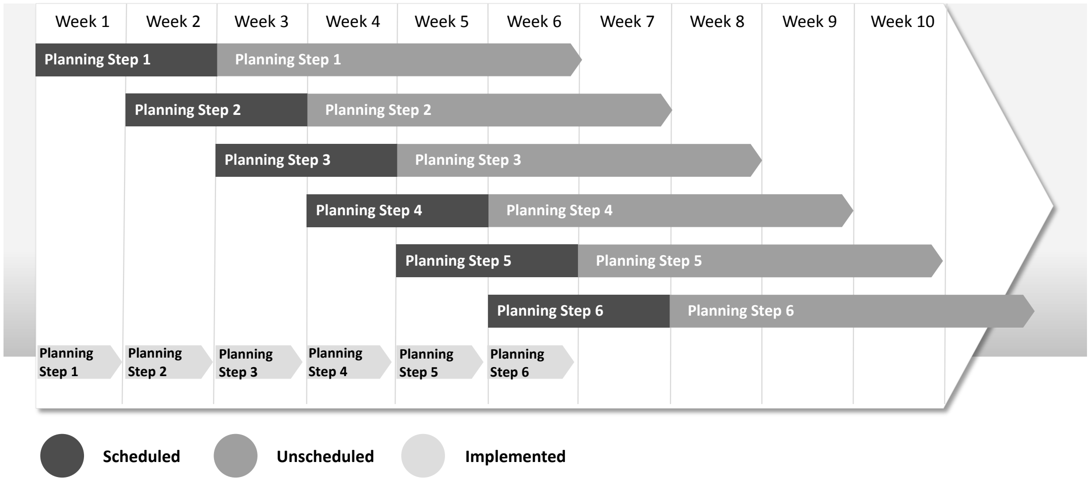

Figure 3: Rolling planning

expected to quickly deteriorate. Moreover, as discussed in the previous section, the need for an exact solution for later periods is questionable on the perspective of a rolling horizon planning. As such and aligned to the frequent re-planning performed by managers, the next sections discuss approximations of this model that can result in less computational expensive formulations solvable in a reasonable time for real-world problems.

4. The rolling horizon models

Aiming to reduce the complexity of model CSP, we consider a time-oriented partition of the planning horizon into two sections: scheduled and unscheduled. The scheduled horizon is composed by the first T_s periods of the original horizon $t = 1, \ldots, T_s$, while the remaining periods define the unscheduled horizon $t = T_s + 1, \ldots, T$. At each planning step the scheduled horizon consists of the model CSP presented in Section 3, while for the unscheduled horizon the model is replaced by a simplified version. The idea is to save computational time by creating less detailed plans for future periods, but still having an approximation of the future expenses in terms of cost and preventing future capacity bottlenecks. Hereafter, we call the rolling model to the formulation containing CSP on the scheduled horizon and an approximation on the unscheduled horizon.

Considering that planning steps occur with a frequency Δ, only the first Δ periods of the plan are actually implemented. The size of the scheduled horizon is, therefore, forced to be $T_s \geq \Delta$. Defining T_s greater than Δ gives origin to an overlap ($\Phi = T_s - \Delta$) of the consecutive scheduled horizons, in which decisions are reconsidered. This can translate into better plans for the short term as the approximation improves, since the decisions in the Δ horizon consider additional information given by the more detailed solutions in the overlapping time periods. However, extending too much the overlap can have a significant negative impact on the efficiency of the rolling model. Figure 3 depicts an example of the application of the rolling model in 6 consecutive planning steps. The planning horizon is divided into weeks with a total of 6 weeks to be considered at each planning step ($T = 6$). The scheduled horizon is composed by the first 2 weeks (T_s=2) and only the first week is implemented at each planning step ($\Delta = 1$), hence we have 1 week of overlap in the detailed horizon.

Next we detail the different approximations that can be used in the unscheduled horizon. In this context, we also revise some of the approaches followed by the works highlighted in Section 2.

4.1 Linear relaxation

The most straightforward model to apply in the unscheduled horizon is the linear relaxation of CSP over $t = T_s + 1, \ldots, T$. Hence, variables Z_{imt}, T_{ijmt} and Y_{imt} are redefined as:

$$Z \in \{0,1\}, \quad T_{jimt} \in \{0, \ldots, q_{imt}\}, \quad Y \in \{0,1\}, \quad t = 1, \ldots, T_s, \tag{16}$$

$$0 \le Z \le 1, \qquad 0 \le T_{jimt} \le q_{imt}, \quad 0 \le Y \le 1, \quad t = T_s + 1, \ldots, T. \tag{17}$$

The resulting rolling model is denoted as CSP$_{rel}$. Note that the relaxation over the unscheduled horizon still approximates the future costs and capacity constraints. This approximation has already been suggested in Clark and Clark [2000].

Applying this approach on a successive series of planning steps is similar to the fix-and-relax (relax-and-fix) heuristic (Dillenberger et al. [1994], Pochet and Wolsey [2006], Federgruen et al. [2007]). Likewise, this constructive heuristic solves a series of partially relaxed MIP subproblems to construct an initial feasible solution to the original MIP. It starts from the first period of the planning horizon and progressively moves forward fixing the integer variables at their optimal value obtained in previous iterations.

The main difference to the heuristic lies on the planning horizon definition. Successive planning steps always define a horizon of T periods: as the first Δ periods are fixed the horizon is expanded by appending periods in the horizon $t = T + 1, \ldots, t + \Delta$. On the other end, in relax-and-fix the initial horizon is never expanded, hence after the fix step the planning horizon is reduced (in case all decisions are fixed in Δ) corresponding to a reduction of the problem size.

Thus, the relax-and-fix heuristic can be seen as an internal rolling horizon approach to solve the original model CSP at each planning step. Following the same idea, it is also valid to state that the following approximations can be applied in an internal rolling horizon approach to solve instances of CSP (by replacing CSP$_{rel}$), widening the applicability of the new results proposed here.

4.2 Increase processing times

Although model CSP$_{rel}$ results in a less computational demanding model, its size can be further reduced. Clark and Clark [2000], Clark [2003] and Clark [2005] explore the idea of completely ignoring setup decisions in the unscheduled horizon, by increasing processing times to include the loss in setup times. Consider $\widehat{p}_{im}$ to be the increased processing times, the following constraints replace (3) for $t = T_s + 1, \ldots, T$:

$$\sum_{i \in \mathscr{A}_m} \widehat{p}_{im} \cdot X_{imt} \le cap_{mt} + \sum_{o} B^o_{mt} \cdot cap^o_{mt} \qquad \forall m, t = T_s + 1, \ldots, T. \tag{18}$$

Note that (3)-(15) only apply to the scheduled horizon and the objective function (1) also needs to be rewritten to consider setup costs only up to T_s.

The question raised by this approximation is how to increase the processing times to include setup times. A simple technique is to do nothing and set $\widehat{p}_{im}$ equal to p_{im} (Clark [2005]); this will be referred as model CSP$_p$. Alternatively, the loss in setups can be included assuming a lot-for-lot production policy (Clark and Clark [2000]) via:

$$\widehat{p}_{im} = \frac{\overline{st}_{im} + p_{im} \cdot \overline{d}_i}{\overline{d}_i}, \qquad \forall i, m \tag{19}$$

$$\widehat{p}_{im} = p_{im}, \qquad \forall i, m \tag{20}$$

where $\overline{d}_i$ is the average period demand for product i and $\overline{st}_{im}$ is the average time to set up machine m to product i. Average setup times are calculated considering setup times incurred when changing to product i and from product i to account for a non-symmetric setup matrix. This approximation to the unscheduled horizon originates the rolling model CSP$_p^f$.

CSP$_p$ overestimates capacity in the unscheduled horizon by completely ignoring future losses due to setups. On the other hand, CSP$_p^f$ may overestimate and/or underestimate capacity. Overestimations are caused by less frequent production lots in comparison to lot-for-lot and also by changeovers below average, while underestimations result from the production of smaller quantities and/or larger setups when compared to the average.

4.3 Reduce available capacity

An alternative to model capacity loss due to setup times when sequencing decisions are eliminated in the unscheduled horizon is to explicitly reduce the capacity by introducing parameter ST_{mt} defined as the estimated setup time expenditure on machine m in period t. The new capacity constraints are as follows:

$$\sum_{i\in\mathscr{A}_m} p_{im}\cdot X_{imt} + ST_{mt} \leq cap_t + \sum_o B_{mt}^o \cdot cap_{mt}^o \qquad \forall m, t \geq T_s + 1. \tag{21}$$

Again the onus of this model is the estimation of ST_{mt}. Setting it to zero would result in CSP$_p$ again. In his work, Stadtler [2003] proposes to estimate future capacity loss based on the losses observed in the setup decisions fixed in previous periods by his internal rolling heuristic. It is suggested to take the setup time loss mean or to increase the mean of the setup time by the absolute deviation average multiplied by a safety factor. To follow a similar reasoning in an external rolling planning, let us transform ST_{mt} into a decision variable. The following constraints are therefore possible estimations for the future capacity loss:

$$ST_{mt} \geq \begin{cases} CSP_{st}^{mean}: & \sum_{l\leq T_s} \frac{\sum_{i,j\in\mathscr{A}_m} st_{ijm}\cdot T_{ijml}}{Ts} \qquad \forall m, t > T_s, & (22a)\\ CSP_{st}^{max}: & \sum_{i,j\in\mathscr{A}_m} st_{ijm}\cdot T_{ijml} \qquad \forall m, l \leq T_s, t > T_s. & (22b)\end{cases}$$

Model CSP$_{st}^{mean}$ estimates future capacity loss by computing the average setup time per machine in the scheduled time periods (22a), while model CSP$_{st}^{max}$ uses the maximum setup loss in the scheduled horizon to reduce capacity in future time periods (22b).

4.4 Sequence independent setups

A drawback of models CSP$_p$, CSP$_p^f$, CSP$_{st}^{mean}$ and CSP$_{st}^{max}$ is that setup costs are neglected over the scheduled horizon and there is no penalty neither for a high number of setups on a given machine and time period, nor for small sized lots. A modeling technique with potential to overcome these problems is to consider sequence independent setups in the unscheduled horizon by using the setup binary variables previously defined (Y_{imt}).

$$\sum_{i\in\mathscr{A}_m} p_{im}\cdot X_{imt} + \sum_{i\in\mathscr{A}_m} \widehat{st}_{im}\cdot Y_{imt} \leq cap_t + \sum_o B_{mt}^o \cdot cap_{mt}^o \qquad \forall m, t. \tag{23}$$

Constraints (23) replace the capacity constraints of model CSP$_p$ and account for future setup times. Constraints (11) are now applied to the entire planning horizon and the term $\sum_{t,m,i\in\mathscr{A}_m} \widehat{sc}_{im}\cdot Y_{imt}$ is added to the objective function to estimate future setup costs. This rolling model is denoted as CSP$_{ind}$ and is closely related to the models presented by Araujo et al. [2007] and Tiacci and Saetta [2012], although the former work does not account for setup costs and the latter does not consider setup times. Values for the sequence-independent parameter $\widehat{st}_{im}$ can be estimated using mean setup time $\overline{st}_{im}$ introduced to incorporate setup times in processing times. The same formula is valid for $\widehat{sc}_{im}$. An alternative approximation can be obtained by the following expression (Tiacci and Saetta [2012]):

$$\widehat{sc}_{im} = \frac{2/(|\mathscr{A}_m|-1)}{\sum_j(1/sc_{ijm}) + \sum_j(1/sc_{jim})} \tag{24}$$

$$\widehat{st}_{im} = \frac{2/(|\mathscr{A}_m|-1)}{\sum_j(1/st_{ijm}) + \sum_j(1/st_{jim})} \tag{25}$$

which tries to factor the fact that setup costs are part of the objective function and, therefore, lower values of sc_{ijm} are more likely to be used than larger ones. The model considering the use of sequence independent setup parameters estimated by (25) will be referred to as CSP^f_{ind}.

5. An iterative method

The quality of the solutions obtained with the rolling models presented in Sections 4.2 to 4.4 is heavily dependent on the estimation accuracy of the approximate parameters. As pinpointed in Stadtler [2003] when incorporating losses in the capacity due to setups, overestimation can cause too many setups in the scheduled horizon, while underestimation might result in finding no feasible solution at all. Dealing with sequence dependencies further stresses the difficulty of this assessment.

The previous methods presented have in common the fact that the estimation considers average values or is based on values observed in the past which implies that information available in the unscheduled horizon is totally or partially neglected. We propose to improve the estimations accuracy by adapting the approximation of the parameters through an iterative method. The idea is to use the rolling model solution for the unscheduled horizon to refine the estimations.

Suppose our rolling model requires the estimation of the approximate parameter u. Our iterative method considers the phases described in Figure 4 at each planning step.

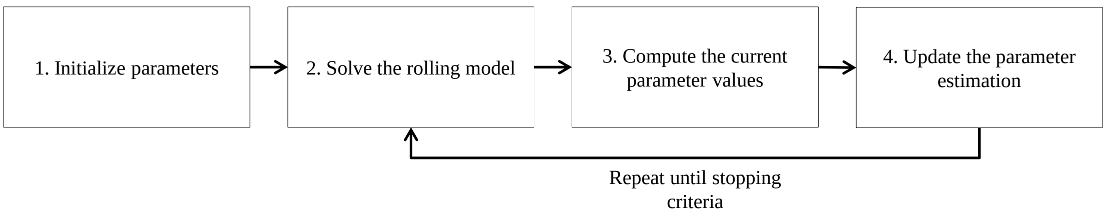

Figure 4: Iterative method for refining approximate parameters

Let r identify the current iteration of the algorithm and u^r the estimation for the approximate parameter available at the end of iteration r (u^0 represents the initial estimation). Our algorithm works as follows. At the beginning of each iteration r the rolling model is solved considering the estimation u^{r-1}. The current values of u^* are computed based on the current solution of the rolling model. Then they are used to update u^r, the values for the next iteration. These steps are repeated until the stopping criteria is met.

Initialize approximate parameters. The initial values of u can be set by any of the techniques introduced in the previous section, e.g. for the estimation of processing times factoring setup time loss both the approaches of CSP_p and CSP_p^f are possible initializations for our method. For models CSP_{st}^{mean} and CSP_{st}^{max} the first approximation is given by (22) which is later updated as explained next.

Compute the current values of the approximate parameters. To compute the values of the approximate parameters it is required to calculate the estimated loss in setups (time and cost) during the unscheduled horizon suggested by the current solution of the rolling model. For this purpose let $\dot{X}_{imt}^r$ define the production quantities defined by the rolling model in iteration r for product i on machine m in periods $t = T_s+1,\ldots,T$. By definition the current setups in the unscheduled horizon are:

$$\dot{Y}_{imt}^r = \begin{cases} 1, & \text{if } \dot{X}_{imt}^r > 0 \quad \forall m, i \in \mathscr{A}_m, t > T_s, \quad (26a) \\ 0, & \text{if } \dot{X}_{imt}^r = 0 \quad \forall m, i \in \mathscr{A}_m, t > T_s. \quad (26b) \end{cases}$$

The non-zero $\dot{Y}_{imt}^r$ determine the products to be sequenced in each period of the unscheduled horizon. The potential sequences can be obtained by solving a Sequential Ordering Problem (SOP) for each machine (see Escudero [1988]). The SOP is a problem related to the Asymmetric Traveling Salesman Problem (ATSP), but where precedence relations between nodes exist, hence it is also called Precedence Constrained Asymmetric Traveling Salesman Problem (PCATSP). A solution to the SOP is a tour passing by all nodes and respecting the precedence constraints. In this context nodes correspond to the non-zero $\dot{Y}_{imt}^r$, arcs are the possible setups between the products to be sequenced and the precedence constraints define that all setups of period t must precede setups of the following periods $t \in \{t+1,\ldots,T\}$.

We can define the SOP arising in this context as follows. A complete directed graph $D = (V,A)$ is given, being V the set of nodes and $A = (\bar{i},\bar{j})|\bar{i},\bar{j} \in V$ the set of arcs. The node set V can be decomposed into $T-T_s$ subsets, one for each period in the unscheduled horizon. Let V_t represent the subset of nodes to be scheduled in period $t \in \{T_s+1,\ldots,T\}$, composed by the products $\bar{i}$ such that $\dot{Y}_{imt}^r = 1$ and for the first and last periods the artificial node 0 is also appended. The arc set is defined by $A = \{(\bar{i},\bar{j})|\bar{i} \in V_t, \bar{j} \in V_t \cup V_{t+1}, t = T_s+1,\ldots,T\}$, arcs between nodes in the same subset represent setups and arcs connecting nodes on the adjacent subset model setup carryover. When traversing an arc $(\bar{i},\bar{j}) \in A$ a cost $c_{\bar{i}\bar{j}} \geq 0$ is incurred, the cost represents the setup cost incurred if the arc connects nodes in the same subset and is zero when connecting nodes of adjacent subsets (carryover). The objective is to find a feasible tour between all scheduled products with the minimal total setup cost. To model the problem as a MIP, let us define for each arc $(\bar{i},\bar{j}) \in A$ a binary variable x_{ij} such that:

$$x_{\bar{i}\bar{j}} = \begin{cases} 1, & \text{if arc } (\bar{i},\bar{j}) \text{ is in the tour,} \quad (27a) \\ 0, & \text{otherwise.} \quad (27b) \end{cases}$$

The MIP formulation for the SOP reads:

$$\text{SOP} \quad \min \sum_{(\bar{i},\bar{j}) \in A} c_{\bar{i}\bar{j}} \cdot x_{\bar{i}\bar{j}} \tag{28}$$

$$\text{s.t.} \quad \sum_{\bar{j} \in V_t} x_{\bar{j}\bar{i}} = 1 \qquad \forall t = T_s+1, \bar{i} \in V_t \tag{29}$$

$$\sum_{\bar{j} \in V_t \cup V_{t-1}} x_{\bar{j}\bar{i}} = 1 \qquad \forall t = T_s+2,\ldots,T, \bar{i} \in V_t \tag{30}$$

$$\sum_{\bar{j} \in V_t \cup V_{t+1}} x_{\bar{i}\bar{j}} = 1 \qquad \forall t = T_s+1,\ldots,T-1, \bar{i} \in V_t \tag{31}$$

$$\sum_{\bar{j} \in V_t} x_{\bar{i}\bar{j}} = 1 \qquad \forall t = T, \bar{i} \in V_t \tag{32}$$

$$\{(\bar{i},\bar{j}) : x_{\bar{i}\bar{j}} = 1\} \text{ does not contain subtours} \tag{33}$$

$$x_{\bar{i}\bar{j}} \in \{0,1\} \qquad \forall (\bar{i},\bar{j}) \in A \tag{34}$$

Constraints (29)-(32) and (34) define the assignment problem relaxation of the SOP, and (33) can be written as the single commodity constraints proposed in Section 3.

Since the size of the SOP arising in the context of our study is considerably small, we have chosen to solve it using a commercial solver of the MIP formulation just presented, although many heuristics are available if we want to speed up the solution process. The solution of the SOP together with $\dot{X}^r_{imt}$ and $\dot{Y}^r_{imt}$ can then be used to update the estimation of the parameters in the rolling models. Let $f(i,t)$ be a function that receives a product i of the original problem and transforms it to the corresponding node of V_t in our SOP problem. This allows us to capture a solution for the CSP as follows:

$$\dot{T}^r_{ijmt} = \begin{cases} x_{f(i,t),f(j,t)} & \text{if } \dot{Y}^r_{imt} \geq 1,\ \dot{Y}^r_{jmt} \geq 1 \text{ and } t > T_s, & (35a) \\ 0, & \text{otherwise.} & (35b) \end{cases}$$

$$\dot{Z}^r_{imt} = \begin{cases} \sum\limits_{\bar{j} \in V_{t-1}} x_{\bar{j},f(i,t)} & \text{if } \dot{Y}^r_{imt} \geq 1 \text{ and } t > T_s + 1, & (36a) \\ 0, & \text{otherwise.} & (36b) \end{cases}$$

The current value for the approximate parameters is then calculated as follows.

Increase processing times. To compute the current approximate processing times we distinguish between three cases: there is a setup on machine m for product i in period t (37a); there is at least one setup in the planning horizon on machine m for product i, but no setup in period t (37b); during the entire planning horizon machine m is never set up for product i. In the first case, the approximate processing time is given by the total time during period t machine m has been occupied to produce product i (setup time + total processing time) divided by the lot size. In the second case, the same idea applies but we use the total production times and quantities, and setup times over the entire planning horizon. Furthermore, in contrast to the previous case, we also impose that at least ϕ percent of product's i total demand has to be produced on machine m. By doing so, we aim to prevent that small lot sizes in a given period have an effect in all the approximate processing times on the machine. Finally, if in the current solution no setup for product i occurs the previous estimation is carried based on (37c):

$$\widehat{p}^*_{imt} = \begin{cases} \left(\sum\limits_j \dot{T}^r_{jimt} \cdot st_{jimt} + p_{im}\dot{X}^r_{imt}\right) / \dot{X}^r_{imt}, & \text{if } \dot{Y}^r_{imt} \geq 1, & (37a) \\ \left(\sum\limits_{t,j} \dot{T}^r_{jimt} \cdot st_{jimt} + p_{im} \cdot \sum\limits_t \dot{X}^r_i\right) / \sum\limits_t \dot{X}^r_i, & \text{if } \dot{Y}^r_{imt} = 0 \text{ and } \sum_t \dot{X}^r_{imt} \geq \phi \cdot \overline{d}_i, & (37b) \\ \widehat{p}^{r-1}_{imt}, & \text{otherwise.} & (37c) \end{cases}$$

Reduce available capacity. The current losses in the setup time at each machine and period are given by:

$$ST^*_{mt} = \sum_{i,j} \dot{T}^r_{ijmt} \cdot st_{ijmt} \quad \forall m, t = T_s + 1, \ldots, T. \tag{38}$$

Sequence independent setups. When computing the sequence independent parameters we also distinguish between the same three cases that appear for computing the approximate processing times. If a setup for product i on machine m in period t exists, the current sequence independent setup times and costs are set by the solution of the SOP (39a). On the other hand, if no setup exists in the period, but at least one occurs during the planning horizon, we compute the sequence independent parameters as the average of the sequence dependent setups over the planning horizon (39b). When no setup takes place on machine m for product i in the entire planning horizon, then the previous estimations correspond to the current values to be considered (39c):

$$\widehat{st}^{*}_{im}(\widehat{sc}^{*}_{im}) = \begin{cases} \sum_{j} \dot{T}^{r}_{ijmt} \cdot st_{ijmt}(sc_{ijmt}), & \text{if } \sum_j \dot{T}^{r}_{ijmt} \geq 1, \quad (39a) \\ \dfrac{\sum_{t,j} \dot{T}^{r}_{ijmt} \cdot st_{ijmt}(sc_{ijmt})}{\sum_{t,j} \dot{T}^{r}_{ijmt}}, & \text{if } \sum_j \dot{T}^{r}_{ijmt} = 0 \text{ and } \sum_{t,j} \dot{T}^{r}_{ijmt} \geq 1, \quad (39b) \\ \widehat{st}^{r-1}_{im}(\widehat{sc}^{r-1}_{im}), & \text{otherwise.} \quad (39c) \end{cases}$$

Update the estimation of the approximate parameters. To update the estimations of the parameters to be used in the next iteration we apply an exponential smoothing technique. The idea is to reduce the effect of volatility in iterations. Consider the following example. Let us assume that we were using model CSP$_{ind}$ and our initial estimation $\widehat{st}^{0}$ was given by (25). In the first iteration after solving CSP$_{ind}$ and when computing the current values for the sequence independent parameters suppose that $\widehat{st}^{*}_{im}$ is much larger than $\widehat{st}^{0}_{im}$. This can point in two directions: either the initial estimation is of poor quality, or the best sequence of the products scheduled in the period corresponds to a high setup time for product i. In the first case we may want to quickly update our estimation, however the high sequence dependent setups may not occur when sequencing the products in the future, and are the result of a myopic schedule using sequence independent setups. The smoothing parameter α^{upd} weighs these two cases according to the following formula:

$$u^{r} = \alpha^{upd} \cdot u^{*} + (1 - \alpha^{upd}) \cdot u^{r-1}. \tag{40}$$

High values of α^{upd} drive a more responsive update, whereas low values of α^{upd} update slowly the estimations.

Stopping criteria. The stopping criteria of the algorithm is defined by calculating the solution's stability in consecutive iterations as a measure of the variation of the estimations. We assume that a stable solution for the rolling model is an indication of the stability of the estimations for the approximate parameters. Following the stability measures suggested in Kimms [1998], we introduce $\dot{q}^{r}_{im}$ as the weighted production quantities associated with product i on machine m in the solution of iteration r:

$$\dot{q}^{r}_{im} = \sum_{t} \xi_{it} \cdot X_{imt} \qquad \forall m, i \in \mathscr{A}, t. \tag{41}$$

The scores of ξ_{it} are defined to weigh more periods in the scheduled horizon since these correspond to the solutions which are to be implemented. A geometric decay function can serve this purpose ($\xi_{it} = t^{-\beta}$). In order to measure the stability over the iterations, we first measure the stability (sm^{r}_{i}) of the production plan defined for each product as follows:

$$sm^{r}_{i} = \frac{\sum_{m|i\in\mathscr{A}} \left|\dot{q}^{r}_{im} - \ddot{q}^{r-1}_{im}\right|}{\max\{\sum_{m|i\in\mathscr{A}} \dot{q}^{r}_{im}, 1\}} \qquad \forall i. \tag{42}$$

We compare the current weighted production quantities with the previous by using $\ddot{q}^{r}_{im}$ which incorporates the past solutions and also relying on exponential smoothing:

$$\ddot{q}^{r}_{im} = \alpha^{stab} \cdot \dot{q}^{r}_{im} + (1 - \alpha^{stab}) \cdot \ddot{q}^{r-1}_{im} \tag{43}$$

The overall solution stability corresponds to the average among all products: $SM^{r} = \frac{1}{N} \cdot \sum_{i} sm^{r}_{i}$. The algorithm stops iterating if $SM^{r} \leq \varepsilon$ and the current solution of the rolling model is implemented.

6. Computational Experiments

In this section we assess the performance of both the rolling models and the iterative approach. The rolling approaches to be compared are summarized in Table 2. We exclude the use of the original CSP model because the preliminary tests conducted, revealed that it is extremely difficult to solve using a commercial solver.

Table 2: Summary of the tested rolling horizon methods

Simplification Strategy	Acronym	Description of the unscheduled model
Linear relaxation	CSP_{rel}	Linear relaxation of model CSP
Increase processing times	CSP_p	Exclude setup times from the capacity constraint (4) of model CSP
	CSP_p^f	Estimation of $\widehat{p}_{im}$ by (20)
Reduce available capacity	CSP_{st}^{mean}	Reduction of the available production line capacity according to the maximum setup loss in the scheduled horizon (22a)
	CSP_{st}^{max}	Reduction of the available production line capacity according to the mean setup loss in the scheduled horizon (22b)
Sequence independent setups	CSP_{ind}^f	Replace sequence dependent setup times (costs) by sequence independent setup times (costs) calculated according to (25)
Iterative method	CSP_p^{iter}	Iterative method on processing times after initial estimation given by CSP_p
	CSP_{st}^{iter}	Iterative method on setup loss after initial estimation given by CSP_{st}^{mean}
	CSP_{ind}^{iter}	Iterative method on sequence indenpendent setup times (costs) after initial estimation given by the minimum setup time (cost) to each product

6.1 Design of the experiment

To simulate use of the rolling models under realistic settings we designed a simulation study that mimics the planning system of many companies in the beverage industry and, in particular, is tailored to the case study that inspired this work (planning system described in Figure 3). In our study, each planning step k covers 6 weeks and there is a weekly re-planning frequency ($\Delta = 1$). At each planning step the generation of demand forecasts is reproduced by using a forecast model which based on sales history defines the forecast for the next planning horizon. The simulation study uses the triple exponential smoothing model, also known as Holt Winter's model, to generate forecasts capturing the trend and seasonality in the demand series. To initialize the forecast method two complete seasonal cycles are used and the smoothing constants are optimized at the beginning of each planning step. The demand forecasts are used as inputs to the rolling model under evaluation which is responsible to provide a feasible solution to the next two weeks and an estimation of stock levels for the remaining planning horizon. As the beverage industry works on a make-to-stock basis we assume that safety stock is carried. The safety stock for each product was defined to be equal to $k \cdot \sigma_{sales}$, where σ_{sales} is the sales' standard deviation and k is set to 1.96. During the implementation of the plan for the next week three situations may occur:

1. *actual demand lower than demand forecast:* in this case the excess stock is carried to the next planning step by reducing demand forecasts starting from the first period until all the extra stock is consumed.

2. *actual demand higher than demand forecast, but forecast error is less than safety stock:* in this case in the next planning step the demand forecasts of the first period are increased by safety stock shortage.

3. *actual demand higher than demand forecast and forecast error higher than safety stock:* in this case stockouts occur. Following the assumption of no backlog, the stockout quantity corresponds to lost sales and the initial period demand forecast is increased by the safety stock.

The quality of the approximations is measured by evaluating (F_1) the cost incurred from the partial solutions implemented on a rolling horizon basis and (F_2) the amount of accumulated lost sales (LS_{it}) . Let the planning step be denoted by $k \in \{1,\ldots,K\}$. When used as a superscript in the decision variables introduced before it represents their value at the end of the planning step. The evaluation functions are then given by:

$$F_1 = \sum_k \left[\sum_{t=1}^{\Delta} \left(\sum_i h_{it} \cdot I_{it}^k + \sum_{m,i,j \in \mathscr{A}_m} sc_{ijm} \cdot T_{ijmt}^k + \sum_{o,m} r_{mt}^o \cdot B_{mt}^{ok} \right) \right], \tag{44}$$

$$F_2 = \sum_k \sum_{t=1}^{\Delta} LS_{it}. \tag{45}$$

At the end of our simulation we are comparing the solutions implemented for 6 consecutive weeks. We assume that at the end of each planning step new data becomes available for $t = T+1,\ldots,T+\Delta$ and that the data between $t = \Delta+1,\ldots,T$ is updated.

We test the models in two scenarios: no forecasting error (forecasts are equal to the demand) and with forecasting errors (forecasts generated by the Holt Winter's model).

All the computational experiments were conducted on Intel 2.40 GHz processing units limited to 4 GB of random access memory and using the Linux operating system. The formulations and the iterative method were implemented in C++ using the ILOG Concert Technology and compiled with gcc compiler. IBM ILOG Cplex 12.4 was used to solve the mixed integer programming models. A limit of 10 minutes is imposed to each planning step, which is considered to be a reasonable waiting time to the managers sponsoring the case study.

6.2 Design of the test bed

The data set is composed of instances randomly generated, but matching the problem features arising in the case study described, which are common to many semi-continuous process industries. The motivation for creating random instances lies on the small number of real-world instances that we were able to collect, but also on the need to extend our analysis to problems with different production environments. The instance design intends to test the influence of the main problem features in the performance of the different rolling models: demand profile, setup variability and tightness of the capacity. This is summarized in Table 3.

Demand Pattern. To simulate the demand pattern in the beverage industry, we have created two profiles: *Stationary* and *Clustered*. The realism is preserved by considering demand seasonality in both profiles. Demand is generated according to the following formula:

$$d_{it} = \mu_i \cdot s(t - \left\lfloor \frac{t}{c} \right\rfloor) + \sigma_i \cdot \delta_t \quad \forall i,t, \tag{46}$$

where μ_i is the average demand of product i, $s(t - \lfloor \frac{t}{c} \rfloor)$ is the seasonal effect on demand (c is the length of the seasonal cycle), σ_i is the standard deviation of product's i demand and δ_t is a normally distributed

Table 3: Levels for the problem features

Problem feature	
Demand profile	Stationary Clustered
Setup times variability	Low High
Tightness of capacity	Tight Very Tight

random variable. Since the planning horizon is relatively short no trend is considered when computing the demand. Seasonality accounts for the end of month effect, which is defined by an increase of sales in the final weeks of each month, therefore in our tests $c = 4$. Because this effect is more evident in some products than others we define three functions for s: no effect $\{1.0, 1.0, 1.0, 1.0\}$, moderate effect $\{0.8, 0.8, 1.1, 1.3\}$ and high effect $\{0.6, 0.8, 1.2, 1.4\}$. The demand profiles affect the mean and standard deviation of products as shown in Tables 4 and 5. In the stationary demand profile all products have similar values for the mean (in this case only B products were considered), while the Clustered profile presents high variability in the demand pattern of products (all four product types are considered). In Clustered profile demand series for products with periodic demand are also defined by the time between orders (TBO), i.e. the average number of weeks between periods with non-null demand. Regarding demand variability we also create three typical demand variability profiles: low (X), medium (Y) and high (Z) variability. The standard deviation of the demand profiles are created according to the coefficients of variation (ϑ) defined in Table 5 and using the following formula:

$$\sigma_i = \mu_i \cdot \sqrt{\vartheta^2 - Var(s)} \tag{47}$$

where $Var(s)$ accounts for the variance of the seasonal effects. The sales history needed to initialize the forecast method (two cycles of 4 periods) plus the 6 planning steps of 6 time periods each require the generation of demand figures for 19 time periods.

Table 4: Average demand generation

Product type	Average demand		% of the total products	
			Stationary	Clustered
High volume (A)	μ_i	U[200,400]	0%	20%
Medium volume (B)	μ_i	U[60,140]	100%	65%
Low volume (C)	μ_i	U[5,20]	0%	10%
Periodic	μ_i TBO	100 2	0%	5%

Setup times variability. We introduce different types of setup matrices to test its influence in the rolling models. As the objective is to try to reproduce as close as possible the features of the case study, setup times can be grouped into two kinds: major and minor setups. Major setups occur when changing between products of different product families and minor setups occur within products of the

Table 5: Demand variability generation

Product type	Demand variability		% of the total products	
			Stationary	Pareto
Low variability (X)	ϑ_i	U[0.05,0.3]	60%-80%	60%-80%
Medium variability (Y)	ϑ_i	U[0.3,0.8]	15%-39%	15%-39%
High variability (Z)	ϑ_i	U[0.8,1]	1%-5%	1%-5%

same family. In instances with low setup time variability minor setups are taken at random from {60, 70, 90}, while major setups from {100, 120, 140}. For the case of high variability, setups between products can take values in {45, 50, ..., 85, 90} (same family) and in {120, 150, ..., 210, 240} (different families). Thus, the second case penalizes more heavily the changeover among the different families.

Tightness of capacity. A feature that has proved to have a strong impact on the performance of solution methods is the tightness of the capacity. Again, to shape random instances similar to the case study, capacity is constant throughout the planning horizon corresponding to the number of production hours available in regular time. Its value is determined for each machine according to:

$$Cap_{mt} = \frac{1}{T \cdot Cut} \cdot \sum_{i \in \mathscr{A}} D_i \cdot p_{im} \cdot \frac{p_{im}}{\sum_l p_{il}} \quad \forall m, t, \tag{48}$$

where D_i is the total demand for product i in the planning horizon which is partially assigned to each machine according to a weighted average considering the processing times. The tightness is defined by *Cut* and is held at two levels: 0.8 (tight) and 0.9 (very tight). The existence of setup times and the seasonal effect can result in insufficient capacity. This can be faced by producing in advance or by using overtime. Random instances have two types of overtime $o \in \{1,2\}$, each corresponding to 15% of the regular capacity.

For each product, a machine independent processing time $\overline{p}_i$ is generated using a uniformly distributed function between 2 and 7. Then, the machine dependent processing times are obtained according to the efficiency of each machine, eff_m, taken from uniform distribution U[0.7,1] implying $p_{im} = \overline{p}_i / eff_m$. Holding costs are selected from the interval U[2,10]. To set the overtime and setup cost we follow the similar approach to Özdamar and Birbil [1998]. Let θ be the average cost per unit of time calculated as follows:

$$\theta = \sum_i \frac{h_i}{\sum_m \left(st_{im}/\overline{d}_i + p_{im} \right) \cdot \frac{p_{im}}{\sum_l p_{il}}} \cdot \frac{D_i}{\sum_j D_j} \cdot ratio. \tag{49}$$

This expression estimates the cost saving of having an additional unit of time to reduce the inventory cost. In the random instances *ratio* is set to 3, setup costs are defined by $sc_{ijm} = \theta \cdot st_{ijm}$ and overtime costs by $r^o_{mt} = \theta \cdot (1-o) \cdot (1.1) \cdot cap^o_{mt}$. Thus overtime type 2 costs 10% more than type 1.

For each combination of the demand profile, setup variability and tightness of the capacity we create instances with 2 and 3 machines, and 15 and 25 products, as the problem size can also have an important effect on the performance of models. Products are associated to machines according to a certain probability τ_{avg} limited by an upper and lower bound. Each machine has a τ_{avg}=40% probability of producing each product and can not produce less than 20% nor over 80% of all products.

We generate 10 instances for each combination of the demand profile (two levels), setup times variability (two levels), tightness of capacity (two levels), number of machines (two levels) and number of products (two levels) giving origin to a total of 320 instances.

6.3 Results with no forecasting error

Figures 5 and 6 summarize the solution quality over all the generated instances when no forecasting error is considered. Figure 5 shows for each of the rolling methods the distribution of the percentage deviation from the best solution (lowest value of F_1 among all methods) by using a box plot. The mean percentage deviation for each method is also depicted by the diamond shape. Figure 6 reproduces the estimated commutative distribution of the deviation given by the percentage of the total instances with a solution within a given deviation.

From Figure 5 we observe that using the linear relaxation on the unscheduled horizon yields lower deviations than incorporating in the processing times the losses from setup times. Model CSP_p^f has the worst performance in terms of deviation which can be explained by the fact that the assumption of a lot-for-lot basis and the mean average setup times are far from being observed in practice. As a consequence this model often underestimates the future capacity by considering too much setup time in each period leading to solutions that have higher holding costs when compared to the best solution. On the other hand, the performance of CSP_p is not far from the performance of CSP_{rel}. In theory such a result is not a surprise since relaxing the integrality constraints on setup variables leads to a model that can produce any item by assigning the smallest possible setup time/cost possible. A common feature to both models is the overestimation of the future capacity which leads to solutions with higher overtime costs and lower holding costs when compared to the best solution. Reducing the future capacity based on the setups witnessed in the scheduled horizon outperforms the use of the relaxation, with the use of the maximum loss being better than the use of the average loss. In these models either overestimation or underestimation occur less frequently leading to smaller deviations in terms of overtime and holding costs. An interesting fact is that the use of a more detailed model, CSP_{ind}^f, does not seem to constitute an advantage over CSP_{st}^{mean} and CSP_{st}^{max}. Indeed the worst case performance deteriorates, although the results are very similar to the previous two models, with a slight reduction of setup costs and increase of both overtime and holding costs. The iterative method when applied to either CSP_p, CSP_{st} or CSP_{ind} achieves very interesting results. CSP_p^{iter} clearly outperforms the single point estimations of CSP_p and CSP_p^f, as well as CSP_{st}^{iter} outperforms CSP_{st}^{mean} and CSP_{st}^{max}, and $\mathrm{CSP}_{ind}^{iter}$ is generally better than CSP_{ind}^f.

In Figure 6 on the horizontal axis we have the percentage deviation from the best solution and on the vertical axis the percentage of the total problems, thus each line approximately describes the cumulative distribution of the solutions within a given percentage deviation for each method. We observe that the relative performance of the models is relatively stable with respect to the deviation from the best solution. Focusing first on the percentage of solutions within up to 3% of the best solution, models CSP_p, CSP_p^f and CSP_{rel} have similar performances at this point with the relaxation being slightly better. The main difference between CSP_p^f and the other two models is the fact that few solutions have high quality and the convergence as the deviation is increased is much slower. The second group of models with similar performance are CSP_{st}^{mean}, CSP_{st}^{max} and CSP_{ind}^f. The model CSP_{st}^{max} is the best among these three. Interesting is the fact that CSP_{ind}^f performance can be explained by its variability. Low values are found within smaller deviation ranges and there are also some instances in which the model performance is poor. Finally, the three methods using the iterative approach form the third group with similar performance. The chart also points to the fact that both CSP_{st}^{iter} and $\mathrm{CSP}_{ind}^{iter}$ are better than CSP_p^{iter}. Only approximately 70% of the solutions of CSP_p^{iter} are below the 3% deviation in comparison to 75% of CSP_{st}^{iter} and 80% of $\mathrm{CSP}_{ind}^{iter}$.

All these results were obtained with the limit of 10 minutes for each planning step, but most models

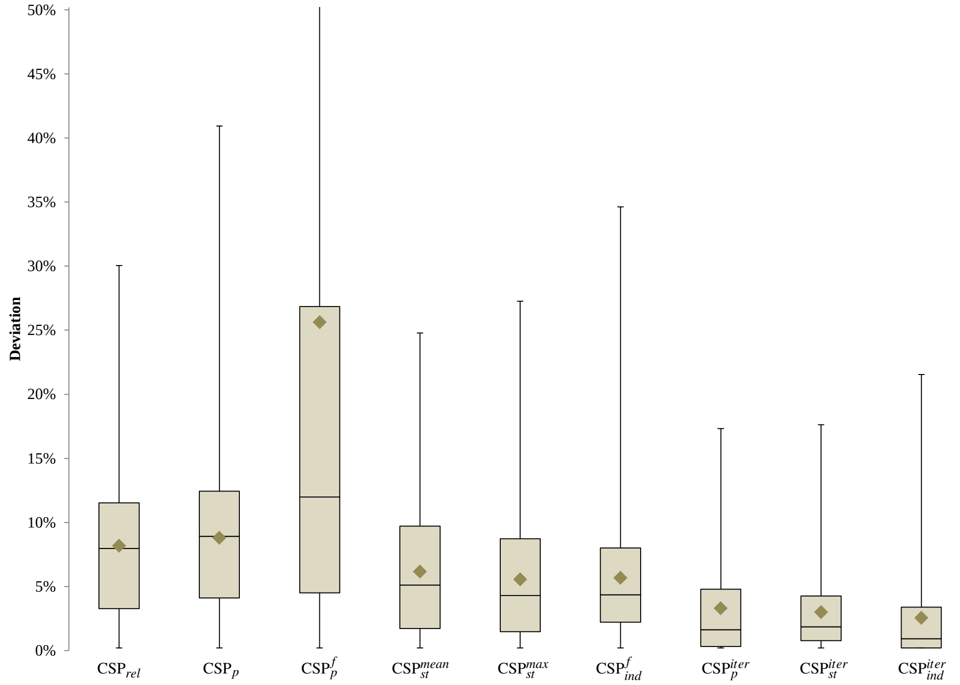

Figure 5: Distribution of the percentage deviation from the best solution for each method with no forecasting error

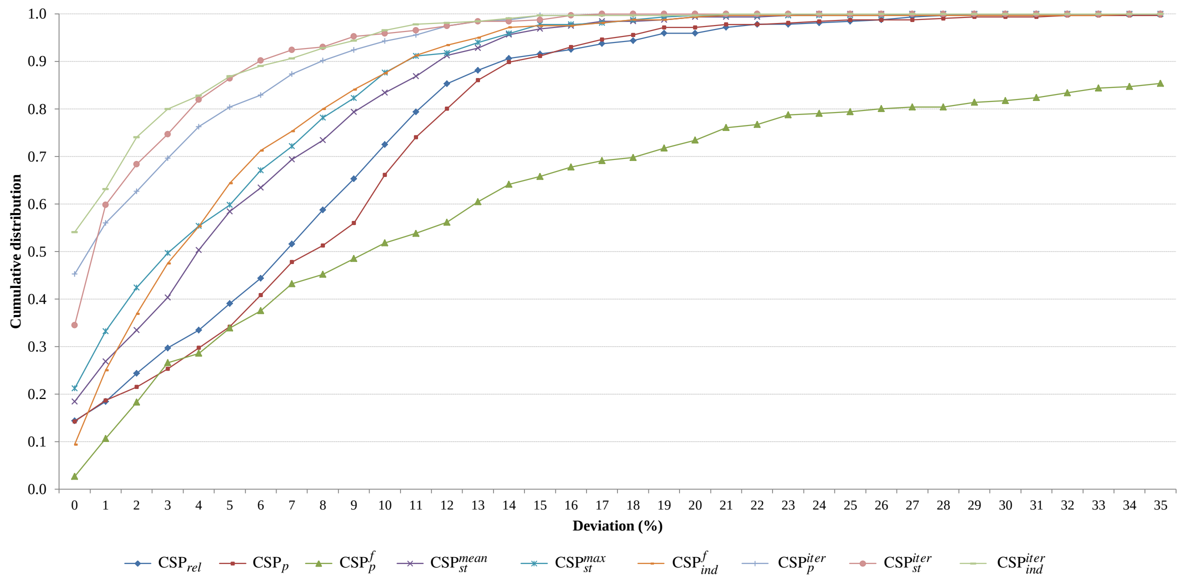

Figure 6: Commulative distribution of the percentage deviation from the best solution for each method with no forecasting error

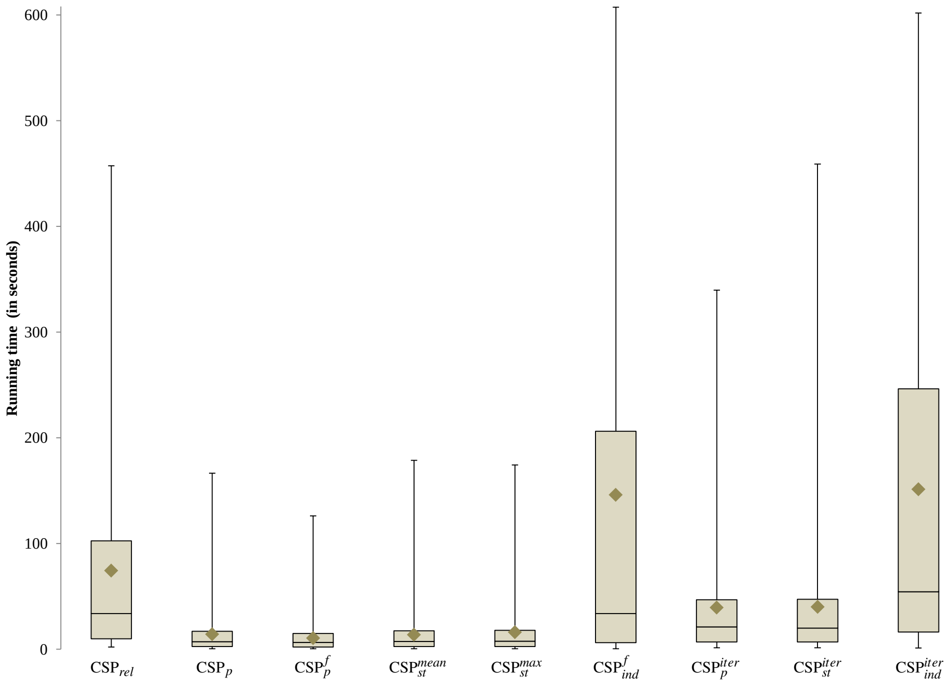

Figure 7: Distribution of the running time for each method with no forecasting error

finished before this threshold. Hence, it is also important to look into the efficiency of the approaches. The iterative approach is expected to take longer running times as the base models have to be solved (re-solved) more than once in each planning step, in comparison to the models that only require one solution of the MIP. Figure 7 depicts the distribution of the running times for each planning step using again a box plot with the diamond denoting the average. As expected the iterative approaches have increased running times. The most computational efficient methods are CSP_p, CSP_p^f, CSP_{st}^{mean} and CSP_{st}^{max}, with a neglectable difference among them. Applying the iterative approach to these models increases the average running time by about 3 times per planning step. Due to the high number of decisions variables in the formulation used in the unscheduled horizon on model CSP_{rel}, this method takes longer than most of the others with the exception of the ones using the sequence independent setups. These are by far the least computational efficient methods, which is not a surprise as the MIP model behind them is substantially harder to solve as a result of the increased number of binary variables. Interestingly applying the iterative approach on CSP_{ind} only slightly increases the average running time of each planning step. This is explained by the fact that after having the optimal solution for the MIP in the first iteration the following re-optimization is carried rather quickly.

Considering both the deviation and running times, the results indicate that CSP_p trades-off better the solution quality versus running time than CSP_{rel}. As suggested by the results, reducing the capacity in the unscheduled periods is preferable over increasing processing times, since for similar running times deviations are lower. In the case of sequence dependent setups, the iterative approach appears as an interesting solution to improve deviation with limited impact on the running times. For the case of CSP_p and CSP_{st}, applying the iterative approach is clearly a question of compromise between solution quality and running time. Overall, and considering that for each planning step a limit of 10 minutes is given, CSP_{st}^{iter} provided the most cost-time efficient solutions.

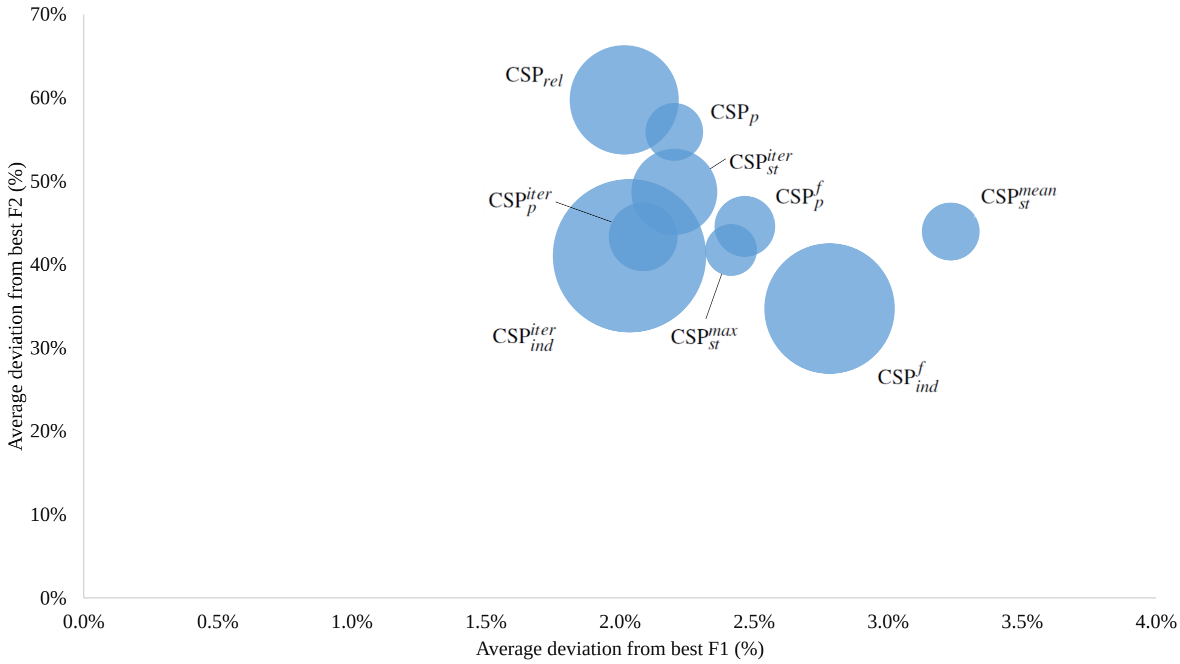

Figure 8: Iterative method for refining approximate parameters

6.4 Results with no forecasting error

Figure 8 summarizes the results of the rolling methods for all the generated instances when introducing forecasting error. The horizontal axis measures the mean percentage deviation from the best solution regarding cost (lowest value of F_1 among all methods), the vertical axis measures the mean percentage deviation from the best solution regarding stockouts (lowest value of F_2 among all methods) and the size of each circle accounts for the running time the model.

The spread of the circles shows that the methods trade-off solution costs and stockouts differently. As when no forecasting error is considered using the linear relaxation on the unscheduled horizon yields lower values for F_1 than incorporating in the processing times the losses from setup times. However, the more conservative estimations of model CSP_p^f, which force the existence of slack in capacity in the periods of the unscheduled horizon, substantially lowers the stockouts (F_2). In fact, both the relaxation and the CSP_p provided the highest levels of stockouts while CSP_p^f is closer to the remaining rolling methods. Reducing the capacity in unscheduled horizon based on the setups witnessed in the scheduled horizon outperforms the use of the previous analyzed models with respect to the stockout by creating additional flexibility, however it yields an increase in setup, holding and overtime costs. For the case of CSP_{st}^{max} the increase in costs is less relevant when compared to CSP_{st}^{mean}. Thus, similarly to the case of no forecasting error the maximum loss being is globally better than the use of the average loss both in terms of costs and stockouts. In contrast to the findings of the study with no forecasting error, detailing the use of capacity performs differently to CSP_{st}^{mean} and CSP_{st}^{max}. CSP_{ind}^f method shows the best performance in terms stockouts. The penalty for the extra setups led the method to keep additional stock that is used to recover from forecasting deficits. The iterative method when applied to either CSP_p, CSP_{st} or CSP_{ind} proved again its value. In the case of incorporating setup times in processing times the iterative method improved both the solution cost and stockout levels. In the remaining two the iterative method favors cost reduction with a slight increase in terms of stockouts.

As in the previous study, the results were obtained with a 10 minute time limit for each planning step. The different circle sizes show two groups requiring a similar computational effort. Three models stand out as the most computational demanding methods CSP_{rel}, CSP_{ind}^f and CSP_{ind}^{iter}. In the case of

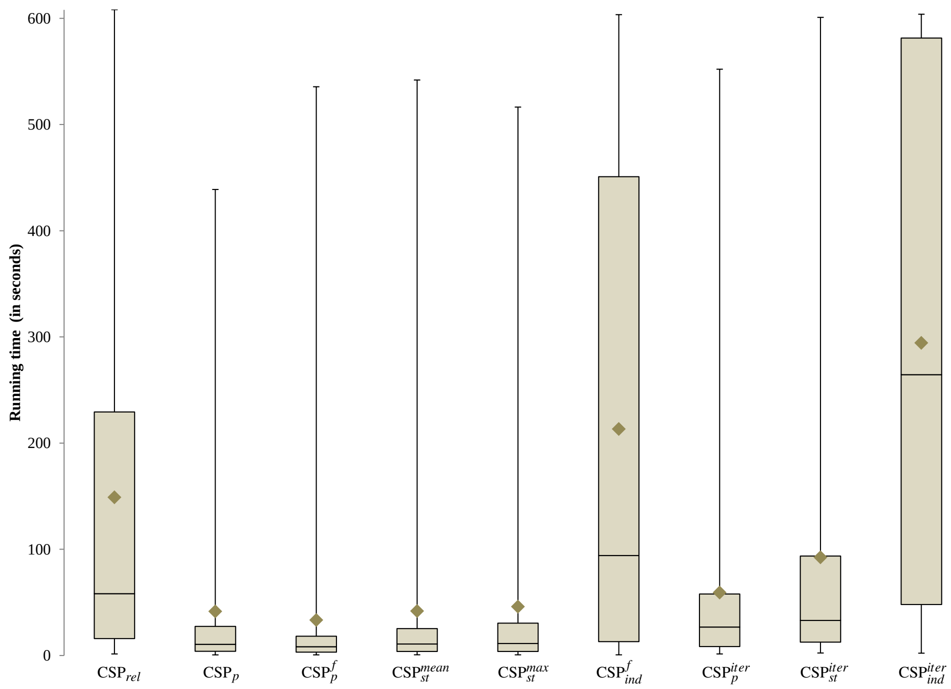

Figure 9: Distribution of the running time for each method with forecasting error

CSP$_{rel}$ the computational burden comes from the model size that has to be solved for the unscheduled horizon whereas for the remaining two methods from the existence of the binary variables. Regarding the other group, the computational effort is much lower and the differences among them are almost neglectable. The only exception comes from the two iterative methods, which despite being competitive are on average slower than the remaining due to their iterative nature. Figure 7 further depicts these differences. Interestingly, we observe that when introducing forecasting error the problem becomes harder for all the methods as their average and worst case performance deteriorates. Forecast errors can create very tight instances with respect to capacity which is known to harden lot sizing problems.

7. Conclusions

In this paper we investigate a rolling horizon approach to the parallel machine lot sizing and scheduling problem with sequence dependent setups. The objective of this work is to render mathematical formulations and methods that can be used in practice to solve the aggregate problem, bearing in mind the rolling planning approach and the constant event-based re-planning. Our study is motivated by the planning problems arising at a case study in the beverage industry.

We follow a line of research that explores the rolling planning features to reduce the complexity of lot sizing and scheduling MIP models by simplifying the decisions beyond the implementation horizon. This reduces the solving time, enhancing their potential applicability in practice. We focus our attention on the determination of the simplification strategies that produce a better estimation of the future capacity and costs, thus leading to the most cost-efficient solutions to be implemented in a rolling basis. We study the proposed simplifications strategies and also introduce a new approximate model that uses the information on setup time expenditure in detailed time periods to reduce the capacity in the simplified horizon. Moreover, these simplifications are appended to a very flexible and computational efficient model to integrate sequencing decisions in lot sizing and scheduling problems.

An important innovation of our work is the iterative method used to improve the accuracy of the parameters defined for the simplified horizon in the context of approximate models. The new method builds on the idea that the solutions for the simplified horizon contain valuable information to refine these estimations. To the best of our knowledge, prior research has just focused on deriving one shot attempt to calculate them. The method is modular and can be applied in several distinct model formulations as shown here.

In our computational tests we focus on analyzing the performance of the MIP models resulting from the various simplification strategies on a large set of instances that mimic the problems faced by the beverage and similar industries. Instances have a high capacity utilization and often require the use of overtime to face the demand peaks. We simulate the planning system of many companies in the beverage and related industries. The results were measured estimating both the setup, holding and overtime time costs but also the stockou of the implemented solutions on a rolling planning. To gather the effect of forecasting error we perform tests with and with no errors.

Overall, we withdraw the following conclusions. Models that tend to underestimate capacity in the unscheduled horizon have poor performances with no forecasting error but can result in a good trade-off between cost and stockouts in the presence of such errors. Reducing the capacity of future periods according to the setups witnessed in the detailed part of the planning horizon leads to a conservative approach and it should be made on the basis of the maximum loss witnessed. The use of the iterative approach improves the quality of the solutions for all rolling methods in both scenarios. The use of sequence independent setups may also be a very interesting approach mainly for two reasons. First, it results in more reasonable rolling solutions as the setup cost and time incurred when scheduling the production of item limits the number of lots per period. Second, it gives the basis to include some operational constraints such as minimum lot sizes due to the presence of binary decision variables in the simplified horizon. Thus, rolling solutions emerging from this approximation can have more value from the managers perspective. Nevertheless, two an important concerns regarding this approximation model are raised. First, the estimation of sequence independent parameters is critical as shown by the improvement by the iterative method. Second its computational tractability as shown in the computational results, pointing out that efficient solution approaches are required in this context.

An important topic for future research is the use of the simplification strategies and the iterative method to create efficient solution algorithm to the static version of the problem as pointed in Section 4.1 and following the works of Clark [2005] and Araujo et al. [2007].

Acknowledgments

The first author is grateful to the Portuguese Foundation for Science and Technology for awarding him a grant (SFRH/BD/62010/2009).

References

S. A. Araujo, M. N. Arenales, and A. R. Clark. Joint rolling-horizon scheduling of materials processing and lot-sizing with sequence-dependent setups. *Journal of Heuristics*, 13(11):337–358, 2007.

A. R. Clark. Optimization approximations for capacity constrained material requirements planning. *International Journal of Production Economics*, 84(2):115 – 131, 2003.

A. R. Clark. Rolling horizon heuristics for production planning and set-up scheduling with backlogs and error-prone demand forecasts. *Production Planning & Control*, 16(1):81–97, 2005.

A. R. Clark and S. J. Clark. Rolling-horizon lot-sizing when set-up times are sequence-dependent. *International Journal of Production Research*, 38(10):2287–2307, 2000.

C. Dillenberger, L. F. Escudero, A. Wollensak, and W. Zhang. On practical resource allocation for production planning and scheduling with period overlapping setups. *European Journal of Operational Research*, 75(2):275 – 286, 1994.

L. Escudero. An inexact algorithm for the sequential ordering problem. *European Journal of Operational Research*, 37(2):236 – 249, 1988.

A. Federgruen, J. Meissner, and M. Tzur. Progressive interval heuristics for multi-item capacitated lot-sizing problems. *Operations Research*, 55(3):490–502, 2007.

B. Fleischmann and H. Meyr. Planning hierarchy, modeling and advanced planning systems. In S. Graves and A. de Kok, editors, *Supply Chain Management: Design, Coordination and Operation*, volume 11 of *Handbooks in Operations Research and Management Science*, pages 455 – 523. Elsevier, 2003.

B. Fleischmann, H. Meyr, and M. Wagner. Advanced planning. In H. Stadtler and C. Kilger, editors, *Supply Chain Management and Advanced Planning*, pages 81–106. Springer Berlin Heidelberg, 2008.

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

L. Guimarães, D. Klabjan, and B. Almada-Lobo. Modeling lotsizing and scheduling problems with sequence dependent setups. Submmited to the European Journal of Operational Research, 2013a.

L. Guimarães, D. Klabjan, and B. Almada-Lobo. Pricing, relaxing and fixing under lot sizing and scheduling. *European Journal of Operational Research*, (0):–, 2013b. doi: 10.1016/j.ejor.2013.04.030. Available in Press.

K. Haase. Capacitated lot-sizing with sequence dependent setup costs. *OR Spectrum*, 18(1):51–59, 1996.

R. J. James and B. Almada-Lobo. Single and parallel machine capacitated lotsizing and scheduling: New iterative MIP-based neighborhood search heuristics. *Computers & Operations Research*, 38 (12):1816 – 1825, 2011.

S. Kang, K. Malik, and L. J. Thomas. Lotsizing and scheduling on parallel machines with sequence-dependent setup costs. *Management Science*, 45(2):273–289, 1999.

A. Kimms. Stability measures for rolling schedules with applications to capacity expansion planning, master production scheduling, and lot sizing. *Omega*, 26(3):355 – 366, 1998.

A. Kovács, K. N. Brown, and S. A. Tarim. An efficient MIP model for the capacitated lot-sizing and scheduling problem with sequence-dependent setups. *International Journal of Production Economics*, 118(1):282 – 291, 2009.

G. Mateus, M. Ravetti, M. Souza, and T. Valeriano. Capacitated lot sizing and sequence dependent setup scheduling: an iterative approach for integration. *Journal of Scheduling*, 13(3):245–259, 2010.

H. Meyr. Simultaneous lotsizing and scheduling on parallel machines. *European Journal of Operational Research*, 139(2):277 – 292, 2002.

L. Özdamar and S. I. Birbil. Hybrid heuristics for the capacitated lot sizing and loading problem with setup times and overtime decisions. *European Journal of Operational Research*, 110(3):525 – 547, 1998.

M. Pinedo and S. Kreipl. Planning and scheduling in supply chains: An overview of issues in practice. *Production and Operations Management*, 13(1):77–92, 2004.

Y. Pochet and L. A. Wolsey. *Production Planning by Mixed Integer Programming (Springer Series in Operations Research and Financial Engineering)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

P. Porkka, A. P. J. Vepsäläinen, and M. Kuula. Multiperiod production planning carrying over set-up time. *International Journal of Production Research*, 41(6):1133 – 1148, 2003.

H. Stadtler. Multilevel lot sizing with setup times and multiple constrained resources: Internally rolling schedules with lot-sizing windows. *Operations Research*, 51(3):487–502, 2003.

L. Tiacci and S. Saetta. Demand forecasting, lot sizing and scheduling on a rolling horizon basis. *International Journal of Production Economics*, 140(2):803 – 814, 2012.